JUDGE CROTTY

Daniel B. Ravicher (DR1498)
David Garrod (DG6759)
PUBLIC PATENT FOUNDATION, INC.
Benjamin N. Cardozo School of Law
55 Fifth Avenue
New York, New York 10003
Tel.: 212-545-5337
Fax.: 212-591-6038

10 CIV 1834

UNITED STATES DISTRICT COURT
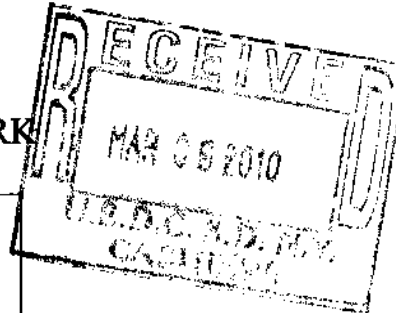SOUTHERN DISTRICT OF NEW YORK

RECEIVED
MAR 05 2010

Public Patent Foundation, Inc.,
a New York, not-for-profit corporation,

                              Plaintiff,

                v.

Adobe Systems Inc.,
a Delaware corporation,

                              Defendant.

Civil Action No. CV _____

COMPLAINT FOR
FALSE PATENT MARKING

Nature of the Action

    1.   This is an action for false patent marking under Title 35, Section 292, of

the United States Code.

    2.   As set forth in detail below, defendant has violated 35 U.S.C. §292(a)

by marking certain products with expired and inapplicable patent numbers, with

the intent of deceiving the public about patent coverage for its products.

3.   Plaintiff seeks an award of monetary damages against defendant, one-half of which shall be paid to the United States pursuant to 35 U.S.C. §292(b).

4.   This Court has subject matter jurisdiction over the present action pursuant to 28 U.S.C. §1338(a). Venue in this Judicial District is proper pursuant to 28 U.S.C. §1338.

The Parties

5.   The Public Patent Foundation, Inc. ("PUBPAT") is a New York, not-for-profit corporation, with a principal place of business located at Benjamin N. Cardozo School of Law, 55 Fifth Avenue, New York, New York 10003.

6.   PUBPAT represents the interests of otherwise unrepresented parties (e.g., consumers, independent software developers, etc.) against various misuses of patents and the patent system by commercial entities.

7.   Upon information and belief, defendant Adobe Systems Inc. ("Defendant") is a Delaware corporation, headquartered in San Jose, CA.

8.   Upon information and belief, Defendant is licensed to do business in the State of New York. (See

http://appsext8.dos.state.ny.us/corp_public/CORPSEARCH.ENTITY_INFORMAT ION?p_nameid=2341754&p_corpid=2299630&p_entity_name=adobe

%20systems&p_name_type=A&p_search_type=BEGINS&p_srch_results_page=0

.)

9.   Upon information and belief, Defendant maintains a regular place of

business at 8 West 40th Street, 8th Floor, New York, NY 10018. (See

http://www.adobe.com/aboutadobe/contact.html.)

10.   Upon information and belief, Defendant sells and distributes software

products throughout the United States, including within this Judicial District.

Defendant's Adobe Reader Product

11.   Upon information and belief, one of the software products that Defendant distributes is known as Adobe Reader.

12.   Upon information and belief, within the period relevant to this action (i.e., the five-year limitations period for §292 claims), Defendant has distributed millions of copies of Adobe Reader to recipients in this Judicial District.

13.   Upon information and belief, Defendant distributes its Acrobat Reader product through, inter alia, Internet transmissions that cause or ultimately result in instances of the Adobe Reader program being installed on the transmission recipients' computers. Upon information and belief, each installed instance of Adobe Reader constitutes a "article" that was "made" by Defendant, as those words appear in 35 U.S.C. §292, because it was Defendant's action(s) that caused the "article" (i.e., the recipient computer with Adobe Reader executable instructions stored therein) to come into existence.
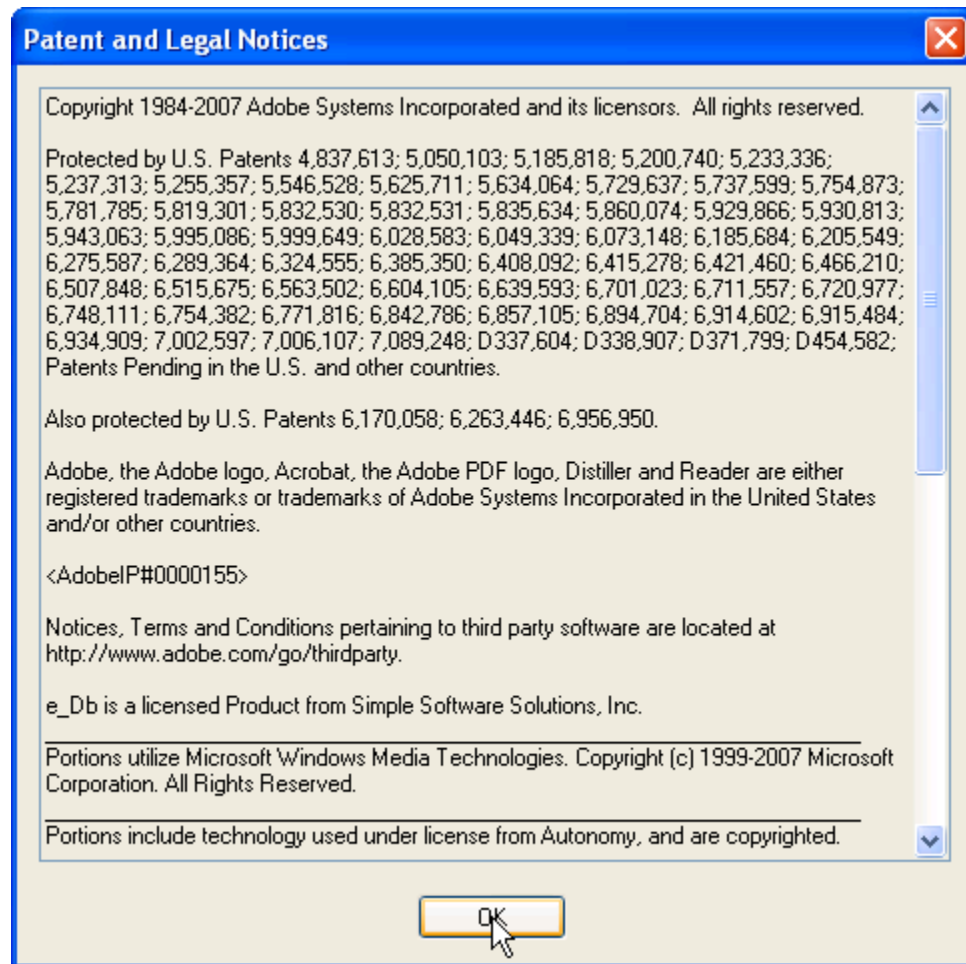
14.   Upon information and belief, starting in early 2005, Defendant began distributing Version 7 of its Reader product.

15.   Upon information and belief, Defendant marked Version 7.05 of its

Adobe Reader product with the following patent notice:

Patent and Legal Notices

Copyright 1984-2004 Adobe Systems Incorporated and its licensors. All rights reserved.

Protected by U.S. Patents 4,667,247; 4,837,613; 5,050,103; 5,185,818; 5,200,740; 5,233,336; 5,237,313; 5,255,357; 5,546,528; 5,625,711; 5,634,064; 5,729,637; 5,737,599; 5,754,873; 5,781,785; 5,819,301; 5,832,530; 5,832,531; 5,835,634; 5,860,074; 5,929,866; 5,930,813; 5,943,063; 5,995,086; 5,999,649; 6,028,583; 6,049,339; 6,073,148; 6,185,684; 6,205,549; 6,275,587; 6,289,364; 6,324,555; 6,385,350; 6,408,092; 6,411,730; 6,415,278; 6,421,460; 6,466,210; 6,507,848; 6,515,675; 6,563,502; 6,604,105 ; 6,639,593; 6,678,410; 6,701,023; 6,711,557; 6,720,977; 6,748,111 ; 6,754,382; 6,771,816; U.S. Patent Design 337,604; 338,907; 371,799; 454,582; Patents Pending.

Adobe, the Adobe logo, Acrobat, the Adobe PDF logo, Distiller and Reader are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

<AdobeIP#0000007>

Size optimization of Adobe Reader performed using FEAD(R) technology licensed from Netopsystems AG, Berlin, Germany. All rights reserved.

Portions include technology used under license from Verity, Inc. and are copyrighted.

Portions utilize Microsoft Windows Media Technologies. Copyright (c) 1999-2002 Microsoft Corporation. All Rights Reserved.

Portions are the result of a cooperative development process by Adobe Systems Incorporated and Microsoft Corporation.

Combined PostScript Drivers are a result of a cooperative development process by Adobe
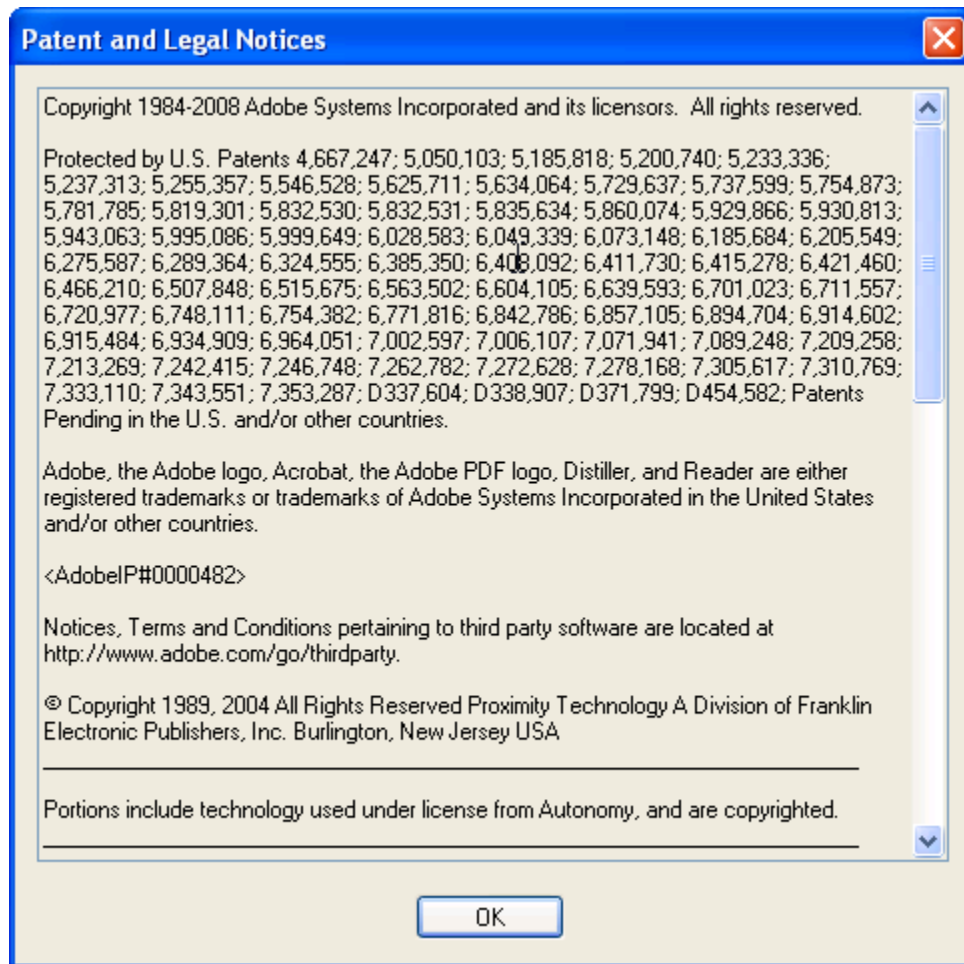
OK

16.   Upon information and belief, starting in late 2006, Defendant began

distributing Version 8 of its Reader product.

17.　Upon information and belief, Defendant marked Version 8.11 of its

Adobe Reader product with the following patent notice:
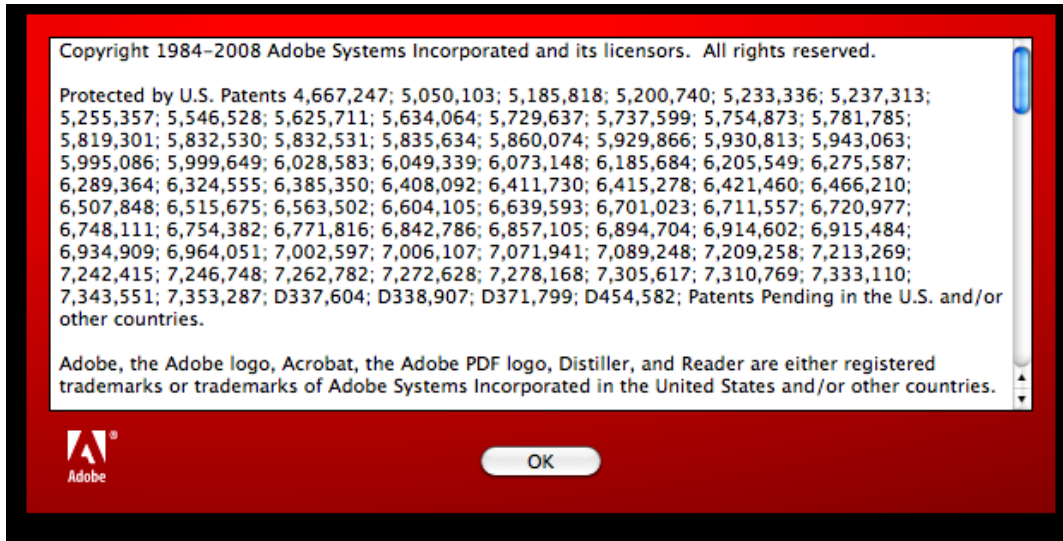

18.　Upon information and belief, starting in mid 2008, Defendant began

distributing Version 9 of its Reader product.

19.   Upon information and belief, Defendant marked Version 9.0 of its

Adobe Reader product with the following patent notice:

20.   Upon information and belief, Defendant marked Version 9.2.0 of its

Adobe Reader product with the following patent notice:

21.   Upon information and belief, Defendant currently distributes version

9.3.1 of Adobe Reader.

22.   Upon information and belief, Defendant began distributing version

9.3.1 or Adobe Reader on Feb. 16, 2010.

23.   Upon information and belief, Defendant marks Version 9.3.1 of its

Adobe Reader product with the following patent notice:
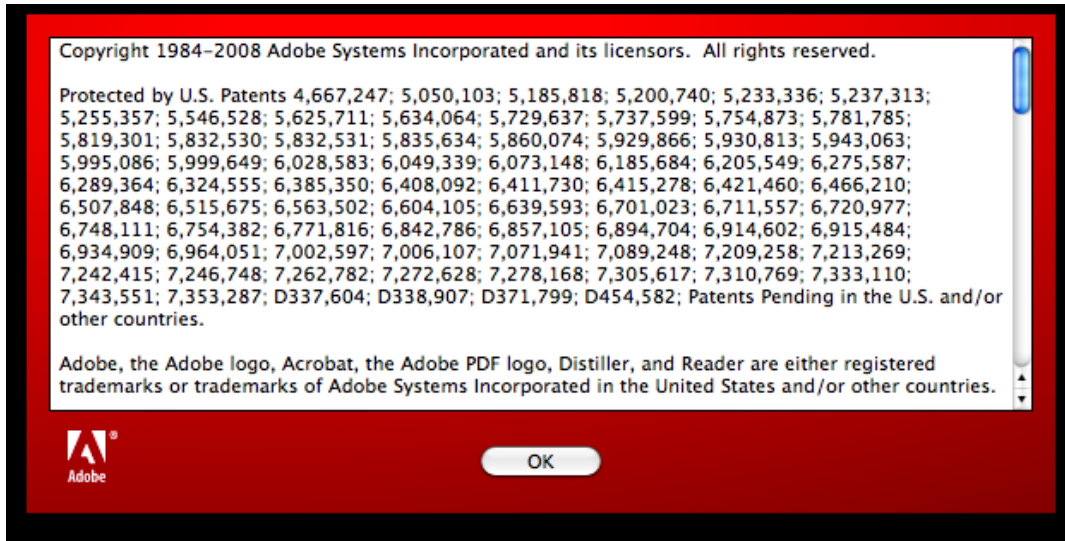
Defendant's Marking of Expired Patents on Adobe Reader

24.   On Dec. 30, 2009, Defendant was sued, under 35 U.S.C. §292, by an

entity named San Francisco Technology Inc. (See San Francisco Technology Inc.

v. Adobe Systems Inc., et al., No. C-09-06083 (N.D. Cal., Dec. 30, 2009) (the "SF

Tech." suit).)

25.   The SF Tech. suit alleges, inter alia, that Defendant violates §292 by

marking Adobe Reader with expired U.S. Patent Nos. 4,837,613; 5,050,103;

D337,604; and D338,907. (See SF Tech., No. C-09-06083 (N.D. Cal., Dec. 30,

2009), ¶¶ 37-42.)

26.   The SF Tech. suit also joins, in the same action, unrelated §292 claims against fourteen other defendants.

27.   Plaintiff does not seek to assert duplicative "expired patent" §292 claims against Defendant in this action. However, plaintiff believes that the SF Tech. suit may be dismissed on a non-merits ground, such as improper joinder under Fed. R. Civ. P. 20.

28.   Accordingly, plaintiff conditionally incorporates and re-alleges herein paragraphs 37-49 from SF Tech.'s Dec. 30, 2009 complaint in the event that the SF Tech. lawsuit is dismissed for improper joinder, lack of standing, or any other non-merits ground.

Defendant's False Marking of Adobe Reader with Inapplicable Patents

29.   In addition to its mis-marking and continued mis-marking (even after the SF Tech. lawsuit) of Adobe Reader with expired patents, Defendant also mis-marks and continues to mis-mark Adobe Reader with the numbers of at least thirty-one unexpired, but inapplicable patents (collectively, the "Inapplicable Patents"), as detailed in paragraphs 30 through 482 below.

30.   U.S. Patent No. 5,185,818 ("the '818 patent"), entitled " METHOD OF

SUBSTITUTING FONTS AND DISPLAYING CHARACTERS," issued on Feb.

9, 1993. (See http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=5185818.PN.&OS=PN/5185818&RS=PN

/5185818.)

31.   The '818 patent contains only one independent claim.

32.   The sole independent claim of the '818 patent reads as follows:

> 1. A method for reconstructing characters in
> a document, which are expressed in one or more
> fonts, to convert said characters to a different font
> designed to closely match characters from a
> plurality of fonts in an aesthetically pleasing
> manner, comprising:
>
> selecting a first character from said document
> determining the width of said first character;
>
> based upon this width determination,
>
> substituting for said first character the same
> character constructed from said different font and
> having the same width as said first character, the
> characters of varying widths within said different
> font being designed to have aesthetically pleasing
> relationships between their parts irrespective of
> width;
>
> continuing the construction and substitution of
> characters of said different font for each of the
> characters in said document needing reconstruction
> in the same manner until all the characters in said

document needing reconstruction have been substituted, thereby creating a reconstructed document having aesthetically pleasing relationship among the characters in said document; and

storing said reconstructed document for subsequent display.

33.   Upon information and belief, claim 1 of the '818 patent does not cover Adobe Reader.

34.   Upon information and belief, Defendant has no reasonable basis to believe that claim 1 of the '818 patent covers Adobe Reader.

35.   Because no independent claim of the '818 patent covers Adobe Reader, the '818 patent does not cover Adobe Reader. Moreover, Defendant lacks any reasonable basis to believe that it does.

36.   U.S. Patent No. 5,625,711 ("the '711 patent"), entitled "METHOD AND APPARATUS FOR PRODUCING A HYBRID DATA STRUCTURE FOR DISPLAYING A RASTER IMAGE," issued on Apr. 29, 1997. (See http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO%2Fsrchnum.htm&r=1&f=G&l=50&s1=5,625,711.PN.&OS=PN/5,625,711&RS=PN/5,625,711.)

37.   The '711 patent contains four independent claims.

38.   The first independent claim of the '711 patent reads as follows:

1. A system for producing a raster image derived from a hybrid data structure including coded and non-coded portions from an input bitmap, the system comprising:

(a) a data processing apparatus;

(b) means for performing recognition on an input bitmap that has been entered into said data processing apparatus to detect identifiable objects within said input bitmap, said means for performing recognition including:

(i) means for comparing each of said identifiable objects with a portion of said input bitmap corresponding to said identifiable object; and

(ii) means for adjusting the size of said identifiable object if said identifiable object is within a threshold size of said corresponding input bitmap portion;

(c) means for creating a hybrid data structure including coded portions corresponding to said identifiable objects and non-coded portions derived from portions of said input bitmap which do not correspond to said identifiable objects; and

(d) an output device for developing a visually perceptible raster image from said hybrid data structure that includes coded images of said identifiable objects and non-coded images of said non-identifiable portions of said input bitmap.

39.   Upon information and belief, claim 1 of the '711 patent does not cover

Adobe Reader.

40.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 1 of the '711 patent covers Adobe Reader.

41.   The second independent claim of the '711 patent reads as follows:

> 3. A method for producing a hybrid data structure from a bitmap of an image including identifiable objects and non-identifiable objects comprising the steps of:
>
> inputting a signal comprising a bitmap of an image into a digital processing apparatus;
>
> determining on said digital processing apparatus a geometric correction of said bitmap, said determining a geometric correction step including creating a distortion map of said bitmap and creating a layout correction transform from said distortion map and said bitmap;
>
> partitioning on said digital processing apparatus said bitmap into lexical units;
>
> assigning on said digital processing apparatus at least one label and an associated confidence level to each lexical unit; and
>
> storing on said digital processing apparatus each lexical unit in a hybrid data structure as an identifiable object if a label for said lexical unit has a confidence level greater than a threshold confidence level, and as a non-identifiable object if no label for said lexical unit has a confidence level greater than said threshold confidence level.

42.   Upon information and belief, claim 3 of the '711 patent does not cover

Adobe Reader.

43.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 3 of the '711 patent covers Adobe Reader.

44.   The third independent claim of the '711 patent reads as follows:

> 4. A method for producing a hybrid image comprising:
>
> inputting a bitmap representing an image into a digital processing apparatus;
>
> segmenting said bitmap into lexical units;
>
> assigning a typeface to each of said lexical units of said bitmap;
>
> recognizing on said digital processing apparatus identifiable objects in said bitmap;
>
> performing a size adjustment to said identifiable objects; and
>
> creating with an output device coupled to said digital processing apparatus a visually perceptible image comprising rendered images of said size-adjusted identifiable objects and bitmap images of objects that were not recognized.

45.   Upon information and belief, claim 4 of the '711 patent does not cover

Adobe Reader.

46.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 4 of the '711 patent covers Adobe Reader.

47.   The final independent claim of the '711 patent reads as follows:

> 5. A method for producing a hybrid data structure from a bitmap of an image including identifiable objects and non-identifiable objects comprising the steps of:
>
> inputting a signal comprising a bitmap of an image into a digital processing apparatus;
>
> partitioning on said digital processing apparatus said bitmap into lexical units, wherein said lexical units are organized into levels of a hierarchy which include a blob level, a character level, a word level, a text line level, a text block level, a page level, and a document level;
>
> assigning on said digital processing apparatus at least one label and an associated confidence level to each lexical unit; and
>
> storing on said digital processing apparatus each lexical unit in a hybrid data structure as an identifiable object if a label for said lexical unit has a confidence level greater than a threshold confidence level, and as a non-identifiable object if no label for said lexical unit has a confidence level greater than said threshold confidence level.

48.   Upon information and belief, claim 5 of the '711 patent does not cover Adobe Reader.

49.   Upon information and belief, Defendant has no reasonable basis to believe that claim 5 of the '711 patent covers Adobe Reader.

50.   Because no independent claim of the '711 patent covers Adobe Reader,

the '711 patent does not cover Adobe Reader. Moreover, Defendant lacks any

reasonable basis to believe that it does.

51.   U.S. Patent No. 5,729,637 ("the '637 patent"), entitled "METHOD

AND APPARATUS FOR PRODUCING A HYBRID DATA STRUCTURE FOR

DISPLAYING A RASTER IMAGE," issued on Mar. 17, 1998. (See

http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=5,729,637.PN.&OS=PN/5,729,637&RS=

PN/5,729,637.)

52.   The '637 patent contains three independent claims.

53.   The first independent claim of the '637 patent reads as follows:

> 1. A system for producing an image
> comprising:
>
> a data processing apparatus;
>
> means for performing recognition on an input
> bitmap that is stored within the data processing
> apparatus to detect objects within the input bitmap
> and to create coded portions therefrom for
> identifiable and non-identifiable objects;
>
> means for creating a data structure including coded
> portions corresponding to identifiable objects and
> links to portions of the input bitmap that

correspond to the identifiable objects and other coded portions corresponding to the non-identifiable objects and links to portions of the input bitmap that correspond to the non-identifiable objects;

an output device for developing a visually perceptible image from the input bitmap, and

means for displaying an editing window and the coded portions of the data structure, wherein the data structure can be changed when displayed in the editing window, and wherein the means for displaying an editing window including a means for comparing each of the coded objects with a portion of the input bitmap corresponding to the coded object and a means for adjusting the size of the coded object when the coded object is within a threshold size of the corresponding input bitmap portion.

54.   Upon information and belief, claim 1 of the '637 patent does not cover Adobe Reader.

55.   Upon information and belief, Defendant has no reasonable basis to believe that claim 1 of the '637 patent covers Adobe Reader.

56.   The second independent claim of the '637 patent reads as follows:

2. A method for producing a data structure from an input bitmap of an image comprising the steps of:

inputting a signal comprising an input bitmap of an image into a digital processing apparatus;

partitioning on the digital processing apparatus the input bitmap into lexical units;

assigning on the digital processing apparatus at least one coded object to each lexical unit;

storing on the digital processing apparatus in a data structure a coded object and link data which links the coded object to its corresponding lexical unit, wherein the bitmap is partitioned into a hierarchy of lexical units, and wherein coded objects are assigned to lexical units having a predetermined hierarchical level and wherein the hierarchy of lexical units includes a blob level, a character level, a word level, a text line level, a text block level, a page level, and a document level.

57.   Upon information and belief, Claim 2 of the '637 patent does not cover Adobe Reader.

58.   Upon information and belief, Defendant has no reasonable basis to believe that claim 2 of the '637 patent covers Adobe Reader.

59.   The final independent claim of the '637 patent reads as follows:

3. A method for producing a data structure from an input bitmap of an image comprising the steps of:

inputting a signal comprising an input bitmap of an image into a digital processing apparatus;

partitioning on said digital processing apparatus the input bitmap into lexical units;

assigning on the digital processing apparatus at

least one coded object to each lexical unit;

storing on the digital processing apparatus in a data
structure a coded object and link data which links
said coded object to its corresponding lexical unit;
and

determining on the digital processing apparatus a
geometric correction of the input bitmap including
creating a distortion map of the input bitmap and
creating a layout correction transform from the
distortion map and the input bitmap.

60.   Upon information and belief, claim 3 of the '637 patent does not cover

Adobe Reader.

61.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 3 of the '637 patent covers Adobe Reader.

62.   Because no independent claim of the '637 patent covers Adobe

Reader, the '637 patent does not cover Adobe Reader. Moreover, Defendant lacks

any reasonable basis to believe that it does.

63.   U.S. Patent No. 5,754,873 ("the '873 patent"), entitled "METHOD

AND APPARATUS FOR SCALING A SELECTED BLOCK OF TEXT TO A

PREFERRED ABSOLUTE TEXT HEIGHT AND SCALING THE REMAINDER

OF THE TEXT PROPORTIONATELY," issued on May 19, 1998. (See

http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=5,754,873.PN.&OS=PN/5,754,873&RS=

PN/5,754,873.)

64.   The '873 patent contains two independent claims.

65.   The first independent claim of the '873 patent reads as follows:

> 1. A computer system for scaling a raster image, the system comprising:
>
> a digital processing apparatus;
>
> a display coupled to said digital processing apparatus for displaying a raster image including text;
>
> a means for retrieving a preferred absolute text height stored by the digital processing apparatus;
>
> a means for selecting a block of text to adjust to the selected preferred absolute text height; and
>
> means for adjusting the entire raster image by an amount which adjusts the selected block of text to the preferred absolute text height and for adjusting the height of the text in the remainder of the raster image proportionately by the same amount.

66.   Upon information and belief, claim 1 of the '873 patent does not cover Adobe Reader.

67.   Upon information and belief, Defendant has no reasonable basis to believe that claim 1 of the '873 patent covers Adobe Reader.

68.   The second independent claim of the '873 patent reads as follows:

4. A method for displaying a preferred size of text on a display device of a computer system, the method comprising:

displaying text having a plurality of heights on a display device;

selecting a subset of the displayed text, all of which has a single text height;

retrieving a different preferred text height stored by the computer system for the selected subset of text;

adjusting the height of all displayed text including the selected subset and the remainder of displayed text by a ratio between the different preferred text height and the single text height; and

re-displaying the text on the display device at the adjusted height.

69.   Upon information and belief, claim 4 of the '873 patent does not cover

Adobe Reader.

70.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 4 of the '873 patent covers Adobe Reader.

71.   Because no independent claim of the '873 patent covers Adobe

Reader, the '873 patent does not cover Adobe Reader. Moreover, Defendant lacks

any reasonable basis to believe that it does.

72.   U.S. Patent No. 5,832,530 ("the '530 patent"), entitled "METHOD

AND APPARATUS FOR IDENTIFYING WORDS DESCRIBED IN A

PORTABLE ELECTRONIC DOCUMENT," issued on Nov. 3, 1998. (See

http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=5,832,530.PN.&OS=PN/5,832,530&RS=

PN/5,832,530.)

73.   The '530 patent contains three independent claims.

74.   The first independent claim of the '530 patent reads as follows:

1. A method for identifying words in a
document comprising:

(a) retrieving a text segment including its x,y
position from a portable electronic document that
has a page including a plurality of characters that
have been identified as characters but not
identified as words and a plurality of text segments
and associated position data;

(b) creating a text object from each text segment
and entering the text object into a linked list of text
objects;

(c) identifying words from the linked list by
analyzing the text object for word breaks and by
analyzing a gap between the text object with a
prior text object using the associated position data;

(d) adding identified words to a word list; and

(e) repeating steps (a) to (e) until the end of the

page is reached.

75.   Upon information and belief, claim 1 of the '530 patent does not cover

Adobe Reader.

76.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 1 of the '530 patent covers Adobe Reader.

77.   The second independent claim of the '530 patent reads as follows:

> 6. A computer program product for
> programming a data processing apparatus to
> identify words on a page of a portable electronic
> document, comprising instructions to:
>
> read from the portable electronic document a
> description of the intended appearance of a page of
> the document when rendered, the description
> including text segments having one or more
> characters to be rendered on the pages each text
> segment associated with position information on
> the page, the position information of the text
> segments defining a display order that is
> independent of a storage order of the text
> segments;
>
> using the position information of the text segments
> to distinguish characters that are part of the same
> word from characters that are not part of the same
> word; and
>
> collect characters that are part of the same word.

78.   Upon information and belief, claim 6 of the '530 patent does not cover

Adobe Reader.

79.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 6 of the '530 patent covers Adobe Reader.

80.   The final independent claim of the '530 patent reads as follows:

> 10. A system for identifying words in a page
> of a portable electronic document, comprising:
>
> a data processing apparatus operable to store a
> page of a portable electronic document as a file
> including a plurality of characters that have not
> been identified as words, wherein each character is
> at least part of a text segment that has associated
> position information indicating where the text
> segment is to be displayed; and
>
> a word identifying program implemented on the
> data processing apparatus for analyzing the text
> segments together with their position data to
> distinguish characters that are part of the same
> word from characters that are not part of the same
> word to create a list of words in the page.

81.   Upon information and belief, claim 10 of the '530 patent does not

cover Adobe Reader.

82.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 10 of the '530 patent covers Adobe Reader.

83.   Because no independent claim of the '530 patent covers Adobe

Reader, the '530 patent does not cover Adobe Reader. Moreover, Defendant lacks

any reasonable basis to believe that it does.

84.   U.S. Patent No. 5,832,531 ("the '531 patent"), entitled "METHOD

AND APPARATUS FOR IDENTIFYING WORDS DESCRIBED IN A PAGE

DESCRIPTION LANGUAGE FILE," issued on Nov. 3, 1998. (See

http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=5,832,531.PN.&OS=PN/5,832,531&RS=

PN/5,832,531.)

85.   The '531 patent contains five independent claims.

86.   The first independent claim of the '531 patent reads as follows:

> 1. A method for identifying words in a page
> description language file containing text characters
> and associated character position data, the
> character position data defining a display order of
> the text characters independent of the storage order
> of the text characters in the file, the method
> comprising:
>
> reading text characters and the associated character
> position data of the text characters from said page
> description language file;
>
> using the character position data to distinguish text
> characters that are part of the same word from text
> characters that are not part of the same word; and
>
> storing text characters that are part of the same
> word into a word buffer.

87.   Upon information and belief, claim 1 of the '531 patent does not cover

Adobe Reader.

88.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 1 of the '531 patent covers Adobe Reader.

89.   The second independent claim of the '531 patent reads as follows:

> 14. A method for detecting words in a
> computer-readable page description language file
> describing the appearance and position of text
> characters, the method comprising:
>
> (a) reading a description including the positions of
> a plurality of text characters from said page
> description language file and storing the text
> characters in a line buffer;
>
> (b) calculating gap and character characteristics of
> text characters in said line buffer from the
> description of the positions of the text characters
> read from the pare description file;
>
> (c) moving a character from said line buffer into a
> word buffer;
>
> (d) repeating steps (c) and (d) until the first of a
> word break and a line buffer empty is detected,
> said word break being detected using said gap and
> character characteristics; and
>
> (e) identifying said characters in said word buffer
> as a word.

90.   Upon information and belief, claim 14 of the '531 patent does not

cover Adobe Reader.

91.   Upon information and belief, Defendant has no reasonable basis to believe that claim 14 of the '531 patent covers Adobe Reader.

92.   The third independent claim of the '531 patent reads as follows:

> 22. A computer-implemented method for searching for a word in a page description language file containing text characters to be displayed or printed and associated character coordinates for each text character or group of text characters, the character coordinates defining a display order of the text characters independent of the storage order of the text characters in the file, the method comprising:
>
> (a) receiving a search word;
>
> (b) processing text characters in said page description language file based upon associated character coordinates of the text characters to create a list of identified words including word coordinates;
>
> (c) comparing said list of identified words to said search word; and
>
> (d) providing the coordinates of words in said list of identified words which match said search word.

93.   Upon information and belief, claim 22 of the '531 patent does not cover Adobe Reader.

94.   Upon information and belief, Defendant has no reasonable basis to believe that claim 22 of the '531 patent covers Adobe Reader.

95.   The fourth independent claim of the '531 patent reads as follows:

> 25. A system for identifying words in a page description language file, comprising:
>
> a data processing apparatus storing a page description language file having characters, each character associated with position data and characteristics data; and
>
> a word identifying program implemented on the data processing apparatus to distinguish text characters that are part of the same word from text characters that are not part of the same word to group the characters into words according to the characteristics data and position data.

96.   Upon information and belief, claim 25 of the '531 patent does not cover Adobe Reader.

97.   Upon information and belief, Defendant has no reasonable basis to believe that claim 25 of the '531 patent covers Adobe Reader.

98.   The final independent claim of the '531 patent reads as follows:

> 30. A system for identifying words in a page description language file, comprising:
>
> a storage device for storing a page description language file having a plurality of text characters, each text character being associated with position data that define a display order of the text characters independent of the storage order of the text characters in the file; and
>
> a data processing apparatus configured by a word

identifying program to:

sort the text characters in the page description language file by their position data, analyze the position data to distinguish text characters that are part of the same word from text characters that are not part of the same word; and

group the characters that are part of the same word.

99.   Upon information and belief, claim 30 of the '531 patent does not cover Adobe Reader.

100.   Upon information and belief, Defendant has no reasonable basis to believe that claim 30 of the '531 patent covers Adobe Reader.

101.   Because no independent claim of the '531 patent covers Adobe Reader, the '531 patent does not cover Adobe Reader. Moreover, Defendant lacks any reasonable basis to believe that it does.

102.   U.S. Patent No. 5,835,634 ("the '634 patent"), entitled "BITMAP COMPARISON APPARATUS AND METHOD USING AN OUTLINE MASK AND DIFFERENTLY WEIGHTED BITS," issued on Nov. 10, 1998. (See http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=5,835,634.PN.&OS=PN/5,835,634&RS=

PN/5,835,634.)

103.   The '634 patent contains four independent claims.

104.   The first independent claim of the '634 patent reads as follows:

> 1. A method for comparing a first and a second bitmap of an image, each bitmap having an outline mask, comprising:
>
> (a) comparing the first and the second bitmaps to produce a difference map of the bits which differ between the two bitmaps;
>
> (b) dividing the difference map of bits into multiple pluralities of bits of differing importance based upon whether they fall on one of the respective outline masks of the first and second bitmaps; and
>
> (c) deriving a comparison score of a match between the two images using the difference map and weighting differently the multiple pluralities of bits of differing importance.

105.   Upon information and belief, claim 1 of the '634 patent does not

cover Adobe Reader.

106.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 1 of the '634 patent covers Adobe Reader.

107.   The second independent claim of the '634 patent reads as follows:

> 2. A computer-implemented method for comparing first and second bitmaps of images, comprising:
>
> (a) obtaining the first and second bitmaps to be

compared;

(b) producing a third bitmap containing any bits which differ between the first and second bitmaps;

(c) obtaining a respective pair of outline masks based upon the first and second bitmaps;

(d) identifying certain bits within the third bitmap to be weighted differently from the remaining bits of the third bitmap based upon whether they fall within one of the respective outline masks; and

(e) determining a comparison score to indicate the extent to which the first and second images differ by differently weighting the certain bits of the third bitmap and the remaining bits.

108.   Upon information and belief, claim 2 of the '634 patent does not cover Adobe Reader.

109.   Upon information and belief, Defendant has no reasonable basis to believe that claim 1 of the '634 patent covers Adobe Reader.

110.   The third independent claim of the '634 patent reads as follows:

6. An apparatus for comparing a first bitmap with a second bitmap, each having an outline mask, comprising:

(a) a comparator for comparing the first and the second bitmaps to produce a difference map of the bits which differ between the two bitmaps;

(b) a divider for dividing the difference map of differing bits into multiple pluralities of bits of

differing importance based upon whether they fall on one of the respective outline masks of the first and second bitmaps; and

(c) a comparison score calculator for deriving a score of a match between the two images using the difference map and weighting differently the multiple pluralities of bits of differing importance.

111.   Upon information and belief, claim 6 of the '634 patent does not cover Adobe Reader.

112.   Upon information and belief, Defendant has no reasonable basis to believe that claim 6 of the '634 patent covers Adobe Reader.

113.   The final independent claim of the '634 patent reads as follows:

7. A computer-implemented method for comparing first and second bitmaps of images, along with their respective outline masks, said method comprising:

(a) producing a difference map of differing bits between the first and second bitmaps;

(b) determining a total number of very different bits in the difference map, the very different bits being the ones that are not common to the respective outline masks;

(c) determining a number of bits lost from the first bitmap;

(d) determining a number of bits gained by the first bitmap;

(e) generating a comparison score of a match
between the bitmaps based upon the results of one
or more of the above steps; and

(f) determining the likelihood the two bitmaps
match based upon the comparison score.

114.   Upon information and belief, claim 7 of the '634 patent does not

cover Adobe Reader.

115.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 7 of the '634 patent covers Adobe Reader.

116.   Because no independent claim of the '634 patent covers Adobe

Reader, the '634 patent does not cover Adobe Reader. Moreover, Defendant lacks

any reasonable basis to believe that it does.

117.   U.S. Patent No. 5,860,074 ("the '074 patent"), entitled "METHOD

AND APPARATUS FOR DISPLAYING AN ELECTRONIC DOCUMENT WITH

TEXT OVER OBJECT," issued on Jan. 12, 1999. (See

http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=5,860,074.PN.&OS=PN/5,860,074&RS=

PN/5,860,074.)

118.   The '074 patent contains nine independent claims.

119.   The first independent claim of the '074 patent reads as follows:

1. A method for displaying on a display device of a computer an electronic document having an object and text specified to appear drawn over the object, the method comprising:

deferring displaying the object in favor of displaying the text;

displaying the text;

then displaying an underneath portion of the object specified to appear underneath the text; and

then displaying the text again, whereby the text is displayed over the object.

120.   Upon information and belief, claim 1 of the '074 patent does not

cover Adobe Reader.

121.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 1 of the '074 patent covers Adobe Reader.

122.   The second independent claim of the '074 patent reads as follows:

2. A method for displaying on a display device of a computer an electronic document having an object and text specified to appear drawn over the object, the method comprising:

deferring displaying the object in favor of displaying the text;

displaying the text;

drawing an underneath portion of the object specified to appear underneath the text into an off-

screen buffer the contents of which are not
displayed;

drawing the text into the of-screen buffer after the
underneath portion has been drawn into the off-
screen buffer; and then

displaying the contents of the off-screen buffer,
whereby the text is displayed over the object.

123.   Upon information and belief, clam 2 of the '074 patent does not cover

Adobe Reader.

124.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 2 of the '074 patent covers Adobe Reader.

125.   The third independent claim of the '074 patent reads as follows:

10. A method for downloading to a computer
and displaying on a display device of the computer
an electronic document having a bitmap image and
text specified to appear drawn over the image,
where the text is all the text specified in the
document as having to appear drawn over the
image, the method comprising:

deferring downloading the image in favor of
downloading the text;

displaying the text;

drawing the image into an off-screen buffer the
contents of which are not displayed;

drawing the text into the off-screen buffer after the
image has been drawn into the off-screen buffer;

and then

> copying the off-screen buffer into a display buffer,
> being a buffer the contents of which are displayed
> on the display device, whereby the text is
> displayed over the image.

126.   Upon information and belief, claim 10 of the '074 patent does not cover Adobe Reader.

127.   Upon information and belief, Defendant has no reasonable basis to believe that claim 10 of the '074 patent covers Adobe Reader.

128.   The fourth independent claim of the '074 patent reads as follows:

> 11. A method for displaying on a display
> device of a computer an electronic document
> having an object and text specified appear drawn
> over the object, the method comprising:
>
> deferring displaying the object in favor of
> displaying the text;
>
> displaying the text;
>
> creating a blocking mask corresponding to the
> displayed appearance of the text; and then
>
> displaying an underneath portion of the object that
> is specified to appear drawn underneath the text
> under control of the blocking mask so that
> displaying the object does not overwrite the
> displayed text, whereby the text is displayed over
> the object.

129.   Upon information and belief, claim 11 of the '074 patent does not

cover Adobe Reader.

130.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 11 of the '074 patent covers Adobe Reader.

131.   The fifth independent claim of the '074 patent reads as follows:

> 13. Apparatus comprising a computer-readable storage medium tangibly embodying computer program instructions for displaying on a display device of a computer an electronic document having an object and text specified to appear drawn over the object, comprising instructions to:
>
> defer displaying the object in favor of displaying the text;
>
> display the text;
>
> then display an underneath portion of the object specified to appear underneath the text; and
>
> then display the text again, whereby the text is displayed over the object.

132.   Upon information and belief, claim 13 of the '074 patent does not

cover Adobe Reader.

133.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 13 of the '074 patent covers Adobe Reader.

134.   The seventh independent claim of the '074 patent reads as follows:

> 14. Apparatus comprising a computer-

readable storage medium tangibly embodying computer program instructions for displaying on a display device of a computer an electronic document having an object and text specified to appear drawn over the object, comprising instructions to:

defer displaying the object in favor of displaying the text;

display the text;

draw an underneath portion of the object specified to appear underneath the text into an off-screen buffer the contents of which are not displayed;

draw the text into the off-screen buffer after the underneath portion has been drawn into the off-screen buffer; and then

display the contents of the off-screen buffer, whereby the text is displayed over the object.

135.   Upon information and belief, claim 14 of the '074 patent does not cover Adobe Reader.

136.   Upon information and belief, Defendant has no reasonable basis to believe that claim 14 of the '074 patent covers Adobe Reader.

137.   The eighth independent claim of the '074 patent reads as follows:

22. Apparatus comprising a computer-readable storage medium tangibly embodying computer program instructions for downloading to a computer and displaying on a display device of the computer an electronic document having a

bitmap image and text specified to appear drawn over the image, where the text is all the text specified in the document as having to appear drawn over the image, comprising instructions to:

defer downloading the image in favor of downloading the text;

display the text;

draw the image into an off-screen buffer the contents of which are not displayed;

draw the text into the off-screen buffer after the image has been drawn into the off-screen buffer; and then

copy the off-screen buffer into a display buffer, being a buffer the contents of which are displayed on the display device, whereby the text is displayed over the image.

138.   Upon information and belief, claim 22 of the '074 patent does not cover Adobe Reader.

139.   Upon information and belief, Defendant has no reasonable basis to believe that claim 22 of the '074 patent covers Adobe Reader.

140.   The final independent claim of the '074 patent reads as follows:

23. Apparatus comprising a computer-readable storage medium tangibly embodying computer program instructions for displaying on a display device of a computer an electronic document having an object and text specified to appear drawn over the object, comprising

instructions to:

defer displaying the object in favor of displaying
the text;

display the text;

create a blocking mask corresponding to the
displayed appearance of the text; and then

display an underneath portion of the object that is
specified to appear drawn underneath the text
under control of the blocking mask so that
displaying the object does not overwrite the
displayed text, whereby the text is displayed over
the object.

141.   Upon information and belief, claim 23 of the '074 patent does not

cover Adobe Reader.

142.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 23 of the '074 patent covers Adobe Reader.

143.   Because no independent claim of the '074 patent covers Adobe

Reader, the '074 patent does not cover Adobe Reader. Moreover, Defendant lacks

any reasonable basis to believe that it does.

144.   U.S. Patent No. 5,999,649 ("the '649 patent"), entitled "METHOD

AND APPARATUS FOR PRODUCING A HYBRID DATA STRUCTURE FOR

DISPLAYING A RASTER IMAGE," issued on Dec. 7, 1999. (See

http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=5,999,649.PN.&OS=PN/5,999,649&RS=

PN/5,999,649.)

145.   The '649 patent contains only one independent claim.

146.   The sole independent claim of the '649 patent reads as follows:

> 1. A method for producing a hybrid data structure from an input raster image which has been scanned and converted to an input bitmap, the hybrid data structure including coded portions which represent lexical units contained within a first part of the input bitmap, the lexical units being organized into hierarchical levels selected from the class consisting of a blob level, a character level, a word level, a text line level, a text block level, a page level and a document level, and a non-coded second part of the input bitmap, the coded portions themselves being capable of conversion to bitmap representations of the lexical units, the method comprising:
>
> performing a recognition process on the input bitmap, thereby recognizing the lexical units;
>
> assigning a confidence level to each lexical unit indicating how confidently it has been recognized;
>
> assigning a data code to each lexical unit to which a confidence level has been assigned at or above a predetermined confidence level; and
>
> creating the hybrid data structure including the

assigned data codes, the input bitmap for any
lexical units below the predetermined confidence
level and the non-coded second part of the input
bitmap.

147.   Upon information and belief, claim 1 of the '649 patent does not

cover Adobe Reader.

148.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 1 of the '649 patent covers Adobe Reader.

149.   Because no independent claim of the '649 patent covers Adobe

Reader, the '649 patent does not cover Adobe Reader. Moreover, Defendant lacks

any reasonable basis to believe that it does.

150.   U.S. Patent No. 6,073,148 ("the '148 patent"), entitled

"DISPLAYING ELECTRONIC DOCUMENTS WITH SUBSTITUTE FONTS,"

issued on Jun. 6, 2000. (See http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=6,073,148.PN.&OS=PN/6,073,148&RS=

PN/6,073,148.)

151.   The '148 patent contains two independent claims.

152.   The first independent claim of the '148 patent reads as follows:

1. A method for displaying on a display
device coupled to a computer an electronic
document having text specified to be drawn using

a desired font that is not on the computer, the
method comprising:

initially displaying the text and a region including
the text using a substitute font different from the
desired font;

obtaining the desired font from a source coupled to
the computer for data transmission; and then

redisplaying the text and the region using the
desired font.

153.   Upon information and belief, claim 1 of the '148 patent does not

cover Adobe Reader.

154.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 1 of the '148 patent covers Adobe Reader.

155.   The second independent claim of the '148 patent reads as follows:

7. Apparatus comprising a computer-
readable storage medium tangibly embodying
computer program instructions for displaying on a
display device coupled to a computer an electronic
document having text specified to be drawn using
a desired font that is not on the computer,
comprising instructions to:

display the text and a region including the text
initially using a substitute font different from the
desired font;

obtain the desired font from a source coupled to
the computer for data transmission; and then

redisplay the text and the region using the desired
font.

156.   Upon information and belief, claim 7 of the '148 patent does not

cover Adobe Reader.

157.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 7 of the '148 patent covers Adobe Reader.

158.   Because no independent claim of the '148 patent covers Adobe

Reader, the '148 patent does not cover Adobe Reader. Moreover, Defendant lacks

any reasonable basis to believe that it does.

159.   U.S. Patent No. 6,275,587 ("the '587 patent"), entitled "SECURE

DATA ENCODER AND DECODER," issued on Aug. 14, 2001. (See

http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=6,275,587.PN.&OS=PN/6,275,587&RS=

PN/6,275,587.)

160.   The '587 patent contains five independent claims.

161.   The first independent claim of the '587 patent reads as follows:

> 1. A method of encoding a serial bit stream
> of digital data, the method comprises:
>
> placing a first portion of bits from the input bit
> stream into bit positions of an encoded output unit

in accordance with a first value, said first number
of bits aligned to a bit position in the encoded
output unit in accordance with a second value and
inserting bits of arbitrary value into remaining bit
positions of the unit of the encoded output unit;

placing a remaining portion of bits from the input
bit stream into bit positions of a second encoded
output unit in accordance with a different, first
value, said remaining portion of bits aligned to a
bit position in the second encoded output unit in
accordance with a different, second value and
inserting bits of arbitrary value into remaining bit
positions of the second unit of the encoded output
unit; and

wherein the bits of arbitrary value are noise bits.

162.   Upon information and belief, claim1 of the '587 patent does not

cover Adobe Reader.

163.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 1 of the '587 patent covers Adobe Reader.

164.   The second independent claim of the '587 patent reads as follows:

2. A method of encoding a serial bit stream
of digital data, the method comprises:

placing a first portion of bits from the input bit
stream into bit positions of an encoded output unit
in accordance with a first value, said first number
of bits aligned to a bit position in the encoded
output unit in accordance with a second value and
inserting bits of arbitrary value into remaining bit
positions of the unit of the encoded output unit;

placing a remaining portion of bits from the input
bit stream into bit positions of a second encoded
output unit in accordance with a different, first
value, said remaining portion of bits aligned to a
bit position in the second encoded output unit in
accordance with a different, second value and
inserting bits of arbitrary value into remaining bit
positions of the second unit of the encoded output
unit; and

iteratively generating subsequent first values and
subsequent second values for the actions of
placing, and for each subsequent action of placing,
placing a variable number of bits, as determined by
the subsequent first values from the input bit
stream into subsequent encoded output units.

165.   Upon information and belief, claim 2 of the '587 patent does not

cover Adobe Reader.

166.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 2 of the '587 patent covers Adobe Reader.

167.   The third independent claim of the '587 patent reads as follows:

17. A method of decoding an encoded serial
bit stream, the method comprising:

selecting significant bits from a unit of the encoded
bit stream in accordance with a number of bits
value and a bit alignment value to place the
selected bits in a decoded bit streams;

gathering values of two prime numbers to produce
generated values;

> generating the generated values based on the
> gathered values of the two prime numbers; and
>
> producing the number of bits value and the bit
> alignment value from the generated values.

168.   Upon information and belief, claim 17 of the '587 patent does not

cover Adobe Reader.

169.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 17 of the '587 patent covers Adobe Reader.

170.   The fourth independent claim of the '587 patent reads as follows:

> 29. A computer program product residing on
> a computer readable medium for encoding a serial
> bit stream of digital data, the product comprising
> instructions for causing a processor to:
>
> place a first portion of bits from the input bit
> stream into bit positions of an encoded output unit
> in accordance with a first value, said first number
> of bits aligned to a bit position in the encoded
> output unit in accordance with a second value and
> insert bits of arbitrary value into remaining, bit
> positions of the unit of the encoded output unit;
> and
>
> place a remaining portion of bits from the input bit
> stream into bit positions of a second encoded
> output unit in accordance with a different, first
> value, said remaining portion of bits aligned to a
> bit position in the second encoded output unit in
> accordance with a different, second value and
> insert bits of arbitrary value into remaining bit

positions of the second unit of the encoded output unit; and

iteratively generate subsequent first values and subsequent second values to place a variable number of bits determined by the subsequent first values from the input bit stream into subsequent encoded output units.

171.   Upon information and belief, claim 29 of the '587 patent does not cover Adobe Reader.

172.   Upon information and belief, Defendant has no reasonable basis to believe that claim 29 of the '587 patent covers Adobe Reader.

173.   The final independent claim of the '587 patent reads as follows:

43. A computer program product residing on a computer readable medium for decoding an encoded serial bit stream, the computer program product comprising instructions that cause a processor to:

select significant bits from a unit of the encoded bit stream in accordance with a number of bits value and a bit alignment value to place the selected bits in a decoded bit stream;

gather values of two prime numbers to produce generated values;

generate the generated values based on the gathered values of the two prime numbers; and

produce the number of bits value and the bit alignment value from the generated values.

174.   Upon information and belief, claim 43 of the '587 patent does not

cover Adobe Reader.

175.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 43 of the '587 patent covers Adobe Reader.

176.   Because no independent claim of the '587 patent covers Adobe

Reader, the '587 patent does not cover Adobe Reader. Moreover, Defendant lacks

any reasonable basis to believe that it does.

177.   U.S. Patent No. 6,324,555 ("the '555 patent"), entitled

"COMPARING CONTENTS OF ELECTRONIC DOCUMENTS," issued on

Nov. 27, 2001. (See http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=6,324,555.PN.&OS=PN/6,324,555&RS=

PN/6,324,555.)

178.   The '555 patent contains five independent claims.

179.   The first independent claim of the '555 patent reads as follows:

> 1. A method executed in a computer system
> for comparing electronic documents on a page-by-
> page basis, the method comprising:
>
> storing in a hash table a hash value of page
> attributes of a first document;

using the hash value of a page of the second document to determine whether there is a match of the hash value in the hash table;

pairing the page of the second document with the page of the first document that has the hash value in the hash table;

rendering a bitmap of each of the still unpaired pages of the first and second documents;

storing in a hash table a hash value of the bitmap of each of the still unpaired pages of the first document;

forming hash values of the bitmap of each of the unpaired pages of the second document;

pairing the page of the second document with the page of the first document that has the hash value of the bitmap in the hash table;

storing in a hash table a hash value of a subset of the bitmap of each of the still unpaired pages of the first document;

forming hash values of a subset of the bitmap of each of the still unpaired pages of the second document;

pairing the page of the second document with the page of the first document that has the has value of the subset of the bitmap in the hash table;

pairing a still unpaired page in the first document which immediately follows one page of a page pair, with a still unpaired page in the second document which immediately follows the other

page of the page pair;

pairing a still unpaired page in one document with a blank page in the other document if the unpaired page in the one document immediately following one page of a page pair and if the page in the other document which immediately follows the other page of the page pair is paired; and

highlighting differences between the pages that do not match, on a visual rendering of the pages.

180.   Upon information and belief, claim 1 of the '555 patent does not cover Adobe Reader.

181.   Upon information and belief, Defendant has no reasonable basis to believe that claim 1 of the '555 patent covers Adobe Reader.

182.   The second independent claim of the '555 patent reads as follows:

2. A computer program product for comparing electronic documents, the product residing on a computer-readable medium and comprising instructions for causing a computer to:

store in a hash table a hash value of page attributes of a first document, wherein the page attribute is computed of marking operators of a page;

form hash values of page attributes of a second document;

use the hash value of a page of the second document to determine whether there is a match of the hash value in the hash table;

pair the page of the second document with a page of the first document that has the hash value in the hash table;

render a bitmap of the still unpaired pages:

store in the hash table a hash of the bitmap of each of the still unpaired pages of the first document;

form bitmaps of a selected portion of each of the pages;

store in a hash table of page attributes the hash of the bitmap of the selected portion the page for each of the unmatched pages of the first document;

use hash value of page attributes of the bitmap of the selected portion the page of the second document to determine if the hash value of a page of the second document matches a hash value in the hash table; and

pair a still unpaired page in the first document that immediately follows one page of a page pair, with a still unpaired page in the second document that immediately follows the other page of the page pair.

183. Upon information and belief, claim 2 of the '555 patent does not cover Adobe Reader.

184. Upon information and belief, Defendant has no reasonable basis to believe that claim 2 of the '555 patent covers Adobe Reader.

185. The third independent claim of the '555 patent reads as follows:

3. A method executed in a computer system for comparing electronic documents on a page-by-page basis, the method comprising:

computing a first digest of marking operators of each page of a first and a second document and pairing the pages of the first document with the pages of the second document that have identical first digests;

computing a second digest of a rendered bitmap of each page of the first document that is still unpaired, and of each page of the second document, that is still unpaired, and pairing the still unpaired of the first document with the still unpaired pages of the second document that have identical second digests;

computing a third digest of a subset of the rendered bitmap of each page of the first document that is still unpaired, and of each page of the second document that is still unpaired, and pairing the still unpaired pages of the first document with the still unpaired pages of the second document that have identical third digests;

pairing an unpaired page in the first document which immediately follows a paired page in the first document, with the page in the second document which immediately follows the other of the paired pages in the second document, if the page which immediately follows the other of the paired pages in the second document is also still unpaired;

pairing any still unpaired pages in the first and second document with a blank page inserted in the second and first document; and

highlighting differences between paired pages that do not have identical first digest, on a visual rendering of the paired pages.

186.   Upon information and belief, claim 3 of the '555 patent does not cover Adobe Reader.

187.   Upon information and belief, Defendant has no reasonable basis to believe that claim 3 of the '555 patent covers Adobe Reader.

188.   The fourth independent claim of the '555 patent reads as follows:

4. A method executed in a computer system for comparing electronic documents on a page-by-page basis, the method comprising:

computing a first digest of marking operators of each page of a first and a second document and pairing the pages of the first document with the pages of the second document that have identical first digests:

computing a second digest of a rendered bitmap of each page of the first document that is still unpaired, and of each page of the second document, that is still unpaired, and pairing the still unpaired of the first document with the still unpaired pages of the second document that have identical second digests;

computing a third digest of a subset of the rendered bitmap of each page of the first document that is still unpaired, and of each page of the second document that is still unpaired, and pairing the still unpaired pages of the first document with

the still unpaired pages of the second document
that have identical third digests;

pairing an unpaired page in the first document
which immediately follows a paired page in the
first document, with the page in the second
document which immediately follows the other of
the paired pages in the second document, if the
page which immediately follows the other of the
paired pages in the second document is also still
unpaired;

pairing any still unpaired pages in the first and
second document with a blank page inserted in the
second and first document; and

highlighting differences between paired pages that
do not have identical first digest, on a visual
rendering of the paired pages; and

arranging the paired pages in a difference
document based on an original page sequence in
one of the documents.

189.   Upon information and belief, claim 4 of the '555 patent does not

cover Adobe Reader.

190.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 4 of the '555 patent covers Adobe Reader.

191.   The final independent claim of the '555 patent reads as follows:

5. A method executed in a computer system
for comparing electronic documents on a page-by-
page basis, the method comprising:

computing a first digest of marking operators of each page of a first and a second document and pairing the pages of the first document with the pages of the second document that have identical first digests;

computing a second digest of a rendered bitmap of each page of the first document that is still unpaired, and of each page of the second document, that is still unpaired, and pairing the still unpaired of the first document with the still unpaired pages of the second document that have identical second digests;

computing a third digest of a subset of the rendered bitmap of each page of the first document that is still unpaired, and of each page of the second document that is still unpaired, and pairing the still unpaired pages of the first document with the still unpaired pages of the second document that have identical third digests;

pairing an unpaired page in the first document which immediately follows a paired page in the first document, with the page in the second document which immediately follows the other of the paired pages in the second document, if the page which immediately follows the other of the paired pages in the second document is also still unpaired;

pairing any still unpaired pages in the first and second document with a blank page inserted in the second and first document; and

highlighting differences between paired pages that do not have identical first digest, on a visual rendering of the paired pages; and

arranging in a difference document paired pages
that differ from each other with respect to at least
one of the first, second and third digests.

192.   Upon information and belief, claim 5 of the '555 patent does not

cover Adobe Reader.

193.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 5 of the '555 patent covers Adobe Reader.

194.   Because no independent claim of the '555 patent covers Adobe

Reader, the '555 patent does not cover Adobe Reader. Moreover, Defendant lacks

any reasonable basis to believe that it does.

195.   U.S. Patent No. 6,408,092 ("the '092 patent"), entitled

"HANDWRITTEN INPUT IN A RESTRICTED AREA," issued on Jun. 18, 2002.

(See http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=6,408,092.PN.&OS=PN/6,408,092&RS=

PN/6,408,092.)

196.   The '092 patent contains two independent claims.

197.   The first independent claim of the '092 patent reads as follows:

1. A method for processing handwritten
input received as marks made on a machine-
readable tablet, comprising:

providing a plurality of predefined regions for input on the tablet;

receiving from a user handwritten marks made in a current region of the plurality of predefined regions;

receiving from the user an indication that the marks in the current region are complete and thereupon providing a different one of the plurality of predefined regions as the current region in which to receive handwritten marks made by the user, whereby completed marks are received in a sequence of at least two regions;

repeating the acts of receiving marks and providing a different region until the user indicates that the handwritten input is complete, wherein a region can be reused as a current region without overwriting marks previously made by the user in the region when it was previously the current region; and

representing the handwritten input as a concatenation of the completed marks as written in the sequence of regions, the completed marks being concatenated in a writing direction, the completed marks having positions with respect to each other defined by a logical concatenation of their respective regions in the sequence in which regions were marked.

198.   Upon information and belief, claim 1 of the '092 patent does not

cover Adobe Reader.

199.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 1 of the '092 patent covers Adobe Reader.

200.   The second independent claim of the '092 patent reads as follows:

31. Apparatus comprising a storage medium tangibly embodying program instructions operable for causing a programmable processor to:

provide a plurality of predefined regions for input on the tablet;

receive from a user handwritten marks made in a current region of the plurality of predefined regions;

receive from the user an indication that the marks in the current region are complete and thereupon provide a different one of the plurality of predefined regions as the current region in which to receive handwritten marks made by the user, whereby completed marks are received in a sequence of at least two regions;

repeat the acts of receiving marks and providing a different region until the user indicates that the handwritten input is complete, wherein a region can be reused as a current region without overwriting marks previously made by the user in the region when it was previously the current region; and

represent the handwritten input as a concatenation of the completed marks as written in the sequence of regions, the completed marks being concatenated in a writing direction, the completed marks having positions with respect to each other defined by a logical concatenation of their respective regions in the sequence in which

regions were marked.

201.   Upon information and belief, claim 31 of the '092 patent does not

cover Adobe Reader.

202.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 31 of the '092 patent covers Adobe Reader.

203.   Because no independent claim of the '092 patent covers Adobe

Reader, the '092 patent does not cover Adobe Reader. Moreover, Defendant lacks

any reasonable basis to believe that it does.

204.   U.S. Patent No. 6,411,730 ("the '730 patent"), entitled

"HISTOGRAM FOR GENERATING A PALETTE OF COLORS," issued on Jun.

25, 2002. (See http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=6,411,730.PN.&OS=PN/6,411,730&RS=

PN/6,411,730.)

205.   The '730 patent contains three independent claims.

206.   The first independent claim of the '730 patent reads as follows:

> 1. A method of generating a histogram for a
> collection of colors in which any color may occur
> zero, one, or more times, the colors being selected
> from a range of color values in a color space, the
> method comprising:

dividing the range of color values using a first set of divisions to define a first set of disjoint, uniform volumes in the color space;

providing a first list consisting essentially of first generation entries, each first generation entry being associated with exactly one distinct volume of the first set of volumes;

storing in an entry frequency data representing the frequency with which a color value within the associated volume occurs in the collection of colors;

dividing the range of color values using a second set of divisions to define a second set of disjoint, uniform volumes in the color space; and

creating a second list from the first list, the second list consisting essentially of second generation entries, each second generation entry being associated with exactly one distinct volume of the second set of volumes and storing combined frequency data for color values in the associated volume.

207.   Upon information and belief, claim 1 of the '730 patent does not cover Adobe Reader.

208.   Upon information and belief, Defendant has no reasonable basis to believe that claim 1 of the '730 patent covers Adobe Reader.

209.   The second independent claim of the '730 patent reads as follows:

9. A computer program on a computer-readable medium for generating a data structure

representing the frequency with which colors occur in a collection of colors in which any color may occur zero, one, or more times, the colors being selected from a range of color values in a color space, the program comprising instructions to:

divide the range of color values using an first set of divisions to define a first set of disjoint, uniform volumes in the color space;

provide a first list consisting essentially of first generation entries, each first generation entry being associated with exactly one distinct volume in the first set of volumes;

store in an entry frequency data representing the frequency with which a color value within the volume associated with the entry occurs in the collection of colors;

divide the range of color values using a second set of divisions to define a second set of disjoint, uniform volumes in the color space; and

create a second list from the first list, the second list consisting essentially of second generation entries, each second generation entry being associated with exactly one distinct volume of the second set of volumes and storing combined frequency data for color values in the associated volume.

210.   Upon information and belief, claim 9 of the '730 patent does not

cover Adobe Reader.

211.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 9 of the '730 patent covers Adobe Reader.

212.   The final independent claim of the '730 patent reads as follows:

18. A method of generating a histogram for a collection of colors in which any color may occur zero, one, or more times, the colors being selected from a range of color values in a color space, the method comprising:

(a) dividing the color space into a set of disjoint volumes, each volume encompassing one or more colors in the color space;

(b) for each occurrence of a color in the collection of colors, (i) identifying the volume in which the color is found, (ii) providing an entry associated with the identified volume in a list of entries, each entry in the list being associated with exactly one distinct volume and having frequency data representing the frequency with which colors in the associated volume occur in the collection, and (iii) updating the frequency data of the entry;

(c) before all occurrences in the collection have been processed, (i) redividing the color space into a current set of larger, disjoint volumes, each larger volume being the union of two or more smaller volumes of the previously-defined set of volumes, (ii) combining original entries in the list so that each resulting entry in the list is associated with exactly one distinct volume of the current set of volumes and combines the frequency data of the original entries that were combined to form the resulting entry, and (iii) continuing the processing of step (b); and

(d) generating a color frequency histogram from

the newest entries in the list.

213.   Upon information and belief, claim 18 of the '730 patent does not

cover Adobe Reader.

214.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 18 of the '730 patent covers Adobe Reader.

215.   Because no independent claim of the '730 patent covers Adobe

Reader, the '730 patent does not cover Adobe Reader. Moreover, Defendant lacks

any reasonable basis to believe that it does.

216.   U.S. Patent No. 6,415,278 ("the '278 patent"), entitled

"RETRIEVING DOCUMENTS TRANSITIVELY LINKED TO AN INITIAL

DOCUMENT," issued on Jul. 2, 2002. (See http://patft.uspto.gov/netacgi/nph-

Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=6,415,278.PN.&OS=PN/6,415,278&RS=

PN/6,415,278.)

217.   The '278 patent contains four independent claims.

218.   The first independent claim of the '278 patent reads as follows:

> 1. A method for retrieving documents,
> comprising:
>
> retrieving an initial document, the initial document
> being within a directory path; and

retrieving, without user intervention, only those other documents for which there is a transitive link from the initial document to the other documents and for which the method determines that the transitive link includes only documents which are within the directory path the method parsing each retrieved document to find the links in each retrieved document and determining from each such link whether the link is to a document within the same directory path as the initial document.

219.   Upon information and belief, claim 1 of the '278 patent does not cover Adobe Reader.

220.   Upon information and belief, Defendant has no reasonable basis to believe that claim 1 of the '278 patent covers Adobe Reader.

221.   The second independent claim of the '278 patent reads as follows:

5. A computer program product, residing on a computer readable medium, for retrieving documents, comprising instructions for causing a computer to:

retrieve the initial document being within a directory path; and

retrieve, without user intervention only those other documents for which there is a transitive link from the initial document to the other documents and for which the computer program determines that a transitive link includes only those documents which are within the directory path the program parsing each retrieved document to find the links in each retrieved document and determining from each such link whether the link is to a document

within the same directory path as the initial document.

222.   Upon information and belief, claim 5 of the '278 patent does not cover Adobe Reader.

223.   Upon information and belief, Defendant has no reasonable basis to believe that claim 5 of the '278 patent covers Adobe Reader.

224.   The third independent claim of the '278 patent reads as follows:

> 9. A method for retrieving documents, the method comprising:
>
> retrieving an initial document, the initial document being within a directory path; and
>
> retrieving, without user intervention, only those other documents for which there is a transitive link from the initial document to the other documents and for which the method determines, by examining the links in the retrieved documents, that the transitive link includes only documents which are within the directory path.

225.   Upon information and belief, claim 9 of the '278 patent does not cover Adobe Reader.

226.   Upon information and belief, Defendant has no reasonable basis to believe that claim 9 of the '278 patent covers Adobe Reader.

227.   The final independent claim of the '278 patent reads as follows:

> 10. A computer program product, residing

on a computer readable medium, for retrieving documents, comprising instructions for causing a computer to:

retrieve an initial document, the initial document being within a directory path; and

retrieve, without user intervention, only those other documents for which there is a transitive link from the initial document to the other documents and for which the program determines, by examining the links in the retrieved documents, that the transitive link includes only documents which are within the directory path.

228.   Upon information and belief, claim 10 of the '278 patent does not cover Adobe Reader.

229.   Upon information and belief, Defendant has no reasonable basis to believe that claim 10 of the '278 patent covers Adobe Reader.

230.   Because no independent claim of the '278 patent covers Adobe Reader, the '278 patent does not cover Adobe Reader. Moreover, Defendant lacks any reasonable basis to believe that it does.

231.   U.S. Patent No. 6,639,593 ("the '593 patent"), entitled "CONVERTING BITMAP OBJECTS TO POLYGONS," issued on Oct. 28, 2003. (See http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=6,639,593.PN.&OS=PN/6,639,593&RS=

PN/6,639,593.)

232.   The '593 patent contains eight independent claims.

233.   The first independent claim of the '593 patent reads as follows:

> 1. A method for generating a polygon from a bitmap object defined by object pixels, each of the object pixels having edges shared with adjacent pixels, the method comprising: identifying boundary pixels of the bitmap object, a boundary pixel being an object pixel that shares an edge with a non-object pixel or a non-object pixel that shares an edge with an object pixel; for each identified boundary pixel, identifying all of the perimeter edge line segments, a perimeter edge line segment being an edge line segment that separates an object pixel from a non-object pixel; accumulating the identified perimeter edge line segments to define a polygon; and using the polygon to define traps for the object.

234.   Upon information and belief, claim 1 of the '593 patent does not cover Adobe Reader.

235.   Upon information and belief, Defendant has no reasonable basis to believe that claim 1 of the '593 patent covers Adobe Reader.

236.   The second independent claim of the '593 patent reads as follows:

> 2. A computer program tangibly stored on a computer-readable medium for converting a bitmap object to a polygon, the bitmap object having a boundary defining the perimeter of the bitmap object where the boundary includes one or more boundary pixels having one or more edge

line segments shared with non-object pixels that
contribute to the perimeter, the program
comprising: a tracing module including
instructions for causing a computer to trace around
the boundary of the bitmap object and generate a
plurality of chain direction codes where each chain
direction code defines a direction from a boundary
pixel to an adjacent boundary pixel; a polygon
generating module including instructions for
causing the computer to identify for each boundary
pixel one or more edge line segments shared with
non-object pixels that contribute to the perimeter
based on two or more of the chain direction codes
generated by the tracing module and to produce as
output a polygon having an outline defined by the
edge line segments; and a vector-based trapping
module including instructions for causing the
computer to define traps for the polygon.

237.   Upon information and belief, claim 2 of the '593 patent does not

cover Adobe Reader.

238.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 2 of the '593 patent covers Adobe Reader.

239.   The third independent claim of the '593 patent reads as follows:

3. A computer-implemented method of
trapping a bitmap object, comprising: identifying
boundary pixels of the bitmap object, a boundary
pixel being an object pixel that shares an edge with
a non-object pixel or a non-object pixel that shares
an edge with an object pixel; for each identified
boundary pixel, identifying all of the perimeter
edge line segments of the bitmap object, a
perimeter edge line segment being a line segment

that separates adjacent pixels and contributes to the perimeter of the bitmap object; accumulating the identified perimeter edge line segments to define a polygon; and defining traps for the polygon.

240.   Upon information and belief, claim 3 of the '593 patent does not cover Adobe Reader.

241.   Upon information and belief, Defendant has no reasonable basis to believe that claim 3 of the '593 patent covers Adobe Reader.

242.   The fourth independent claim of the '593 patent reads as follows:

> 5. A computer-implemented method of trapping a bitmap object, the bitmap object having a boundary defining the perimeter of the bitmap object where the boundary includes one or more boundary pixels having one or more edge line segments that contribute to the perimeter, the method comprising: defining a polygon from boundary pixels edge line segments that contribute to a perimeter of the object and accumulating the contributing edge line segments; planarizing a page including the polygon and one or more other vector objects; trapping the polygon against the other vector objects on the page.

243.   Upon information and belief, claim 5 of the '593 patent does not cover Adobe Reader.

244.   Upon information and belief, Defendant has no reasonable basis to believe that claim 5 of the '593 patent covers Adobe Reader.

245.   The fifth independent claim of the '593 patent reads as follows:

6. A method for printing a page including a bitmap object, comprising: defining the bitmap object as a polygon, including identifying boundary pixels of the bitmap object, a boundary pixel being an object pixel that shares an edge with a non-object pixel or a non-object pixel that shares an edge with an object pixel, and for each identified boundary pixel identifying all of the perimeter edge line segments of the bitmap object and accumulating the identified perimeter edge line segments to define the polygon, a perimeter edge line segment being a line segment separating the adjacent pixels and contributing to the perimeter of the bitmap object; mapping the polygon to device space; filling the polygon; and rendering the page using the filled polygon.

246.   Upon information and belief, claim 6 of the '593 patent does not cover Adobe Reader.

247.   Upon information and belief, Defendant has no reasonable basis to believe that claim 6 of the '593 patent covers Adobe Reader.

248.   The sixth independent claim of the '593 patent reads as follows:

7. A method of trapping a bitmap object having a perimeter against a vector object in a vector-based trapping system, comprising: constructing a polygon congruent to the perimeter of the bitmap object, including identifying pixels in the object having one or more edge line segments that contribute to the perimeter and accumulating the edge line segments that contribute to the perimeter; and using the polygon to define vector-based traps for the object.

249.   Upon information and belief, claim 7 of the '593 patent does not cover Adobe Reader.

250.   Upon information and belief, Defendant has no reasonable basis to believe that claim 7 of the '593 patent covers Adobe Reader.

251.   The seventh independent claim of the '593 patent reads as follows:

> 8. A computer program product for printing a page including a bitmap object, the product comprising instructions operable to cause an instruction processor to: receive a bitmap object; define the bitmap object as a polygon by identifying boundary pixels of the bitmap object, a boundary pixel being an object pixel that shares an edge with a non-object pixel or a non-object pixel that shares an edge with an object pixel, and for each identified boundary pixel identifying all of the perimeter edge line segments of the bitmap object and accumulating the identified perimeter edge line segments to define the polygon, a perimeter edge line segment being a line segment separating adjacent pixels and contributing to the perimeter of the bitmap object; map the polygon to device space; fill the polygon; and render the page using the filled polygon.

252.   Upon information and belief, claim 8 of the '593 patent does not cover Adobe Reader.

253.   Upon information and belief, Defendant has no reasonable basis to believe that claim 8 of the '593 patent covers Adobe Reader.

254.   The final independent claim of the '593 patent reads as follows:

10. A computer program product for generating a polygon from a bitmap object defined by object pixels, each of the object pixels having edges shared with adjacent pixels, the product comprising instructions operable to cause an instruction processor to: identifying boundary pixels of the bitmap object, a boundary pixel being an object pixel that shares an edge with a non-object pixel or a non-object pixel that shares an edge with an object pixel; for each identified boundary pixel, identify all of the perimeter edge line segments, a perimeter edge line segment being an edge line segment that separates an object pixel from a non-object pixel; accumulate the identified perimeter edge line segments to define a polygon; and use the polygon to define traps for the object.

255.   Upon information and belief, claim 10 of the '593 patent does not cover Adobe Reader.

256.   Upon information and belief, Defendant has no reasonable basis to believe that claim 10 of the '593 patent covers Adobe Reader.

257.   Because no independent claim of the '593 patent covers Adobe Reader, the '593 patent does not cover Adobe Reader. Moreover, Defendant lacks any reasonable basis to believe that it does.

258.   U.S. Patent No. 6,701,023 ("the '023 patent"), entitled "REDUCING APPEARANCE DIFFERENCES BETWEEN CODED AND NONCODED UNITS OF TEXT," issued on Mar. 2, 2004. (See

http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=6,701,023.PN.&OS=PN/6,701,023&RS=

PN/6,701,023.)

259.   The '023 patent contains four independent claims.

260.   The first independent claim of the '023 patent reads as follows:

> 1. A method for processing text comprising a
> first unit of text and a second unit of text, the
> method comprising: deriving a correction factor
> for the first unit of text, the correction factor being
> derived from a coded representation of the second
> unit of text and an original noncoded
> representation of the second unit of text; and
> modifying an optical density of a representation of
> the first unit of text in accordance with the
> correction factor.

261.   Upon information and belief, claim 1 of the '023 patent does not

cover Adobe Reader.

262.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 1 of the '023 patent covers Adobe Reader.

263.   The second independent claim of the '023 patent reads as follows:

> 12. A computer program product having
> instructions operable to cause a computer to
> process text comprising a first unit of text and a
> second unit of text, the computer program product
> tangibly stored on a computer-readable medium,
> wherein the product comprises instructions
> operable to cause a processor to: derive a

correction factor for the first unit of text, the correction factor being derived from a coded representation of the second unit of text and an original noncoded representation of the second unit of text; and modify an optical density of a representation of the first unit of text in accordance with the correction factor.

264.   Upon information and belief, claim 12 of the '023 patent does not cover Adobe Reader.

265.   Upon information and belief, Defendant has no reasonable basis to believe that claim 12 of the '023 patent covers Adobe Reader.

266.   The third independent claim of the '023 patent reads as follows:

> 23. A system for processing text having a first unit of text and a second unit of text, comprising: means for deriving a correction factor for the first unit of text, the correction factor being derived from a coded representation of the second unit of text and an original noncoded representation of the second unit of text; and means for modifying an optical density of a representation of the first unit of text in accordance with the correction factor.

267.   Upon information and belief, claim 23 of the '023 patent does not cover Adobe Reader.

268.   Upon information and belief, Defendant has no reasonable basis to believe that claim 23 of the '023 patent covers Adobe Reader.

269.   The final independent claim of the '023 patent reads as follows:

30. An electronic document representing text in a page description language, the text having a first unit of text and a second unit of text, wherein a common font typeface is attributed to both the first unit of text and the second unit of text, the electronic document comprising: a first unit of text and a second unit of text, wherein a common font typeface is attributed to both the first unit of text and the second unit of text; a coded representation of the second unit of text in characters of the common font typeface; and a final raster representation of the first unit of text, wherein the final raster representation is a modified representation generated from an original noncoded representation of the first unit of text according to a correction factor, the correction factor being computed from a noncoded representation of the second unit of text and an optical density of a rendered coded representation of the second unit of text.

270.   Upon information and belief, claim 30 of the '023 patent does not cover Adobe Reader.

271.   Upon information and belief, Defendant has no reasonable basis to believe that claim 30 of the '023 patent covers Adobe Reader.

272.   Because no independent claim of the '023 patent covers Adobe Reader, the '023 patent does not cover Adobe Reader. Moreover, Defendant lacks any reasonable basis to believe that it does.

273.   U.S. Patent No. 6,748,111 ("the '111 patent"), entitled "RECOGNIZING TEXT IN A MULTICOLOR IMAGE," issued on Jun. 8, 2004.

(See http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=6,748,111.PN.&OS=PN/6,748,111&RS=

PN/6,748,111.)

274.   The '111 patent contains four independent claims.

275.   The first independent claim of the '111 patent reads as follows:

> 1. A method for identifying and reproducing a background of a pixelmap, comprising: dividing a pixelmap into a grid of tiles; determining for each tile a background component, including comparing the determined background component for a tile to the background component determined for one or more neighboring tiles not including the tile, if the determined background component for the tile does not match the background component determined for one or more of the one or more neighboring tiles, adjusting the background component for the tile to match the background component determined for one of the one or more neighboring tiles; building a low resolution pixelmap representing a background of the pixelmap using the determined background component for each tile; and storing the low resolution pixelmap.

276.   Upon information and belief, claim 1 of the '111 patent does not cover Adobe Reader.

277.   Upon information and belief, Defendant has no reasonable basis to believe that claim 1 of the '111 patent covers Adobe Reader.

278.   The second independent claim of the '111 patent reads as follows:

11. A computer-implemented method for recognizing and reproducing a background in a multicolor image stored in a computer, the method comprising: dividing an image into multiple blocks; analyzing color distributions for each of the blocks; identifying blocks having two main colors; grouping two-color blocks having similar colors into two-color zones; determining a background color for each two-color zone, including comparing the determined background component for a tile to the background component determined for one or more neighboring tiles not including the tile, if the determined background component for the tile does not match the background component determined for one or more of the one or more neighboring tiles, adjusting the background component for the tile to match the background component determined for one or the one or more neighboring tiles; and building a low resolution pixelmap representing the background of the multicolor image using the determined background component for each two-color zone.

279.   Upon information and belief, claim 11 of the '111 patent does not cover Adobe Reader.

280.   Upon information and belief, Defendant has no reasonable basis to believe that claim 11 of the '111 patent covers Adobe Reader.

281.   The third independent claim of the '111 patent reads as follows:

12. A computer-implemented method for processing and reproducing a multicolor image

represented as a pixelmap, the method comprising: dividing a pixelmap into a grid of tiles; classifying each tile as one of either monochrome, picture, or text tiles; assigning a background component to each monochrome tile that is equivalent to a monochrome color associated with a given monochrome tile; determining for each text tile a background component; building a low resolution pixelmap representing a background of the pixelmap using the determined background component for each text tile and the assigned background component for each monochrome tile; processing the text tiles with an optical character recognition process to produce recognized text; and reproducing the multicolor image including rendering directly each picture tile; rendering the low resolution pixelmap, and rendering the recognized text.

282.   Upon information and belief, claim 12 of the '111 patent does not

cover Adobe Reader.

283.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 12 of the '111 patent covers Adobe Reader.

284.   The final independent claim of the '111 patent reads as follows:

14. A computer program product, tangibly stored on a computer-readable medium, for identifying and reproducing a background of a pixelmap, the product comprising instructions operable to cause a programmable processor to: divide a pixelmap into a grid of tiles; determine for each tile a background component, including compare the determined background component for a tile to the background component determined

for one or more neighboring tiles not including the tile, if the determined background component for the tile does not match the background component determined for one or more of the one or more neighboring tiles, adjust the background component for the tile to match the background component determined for one of the one or more neighboring tiles; build a low resolution pixelmap representing a background of the pixelmap using the determined background component for each tile; and store the low resolution pixelmap.

285.   Upon information and belief, claim 14 of the '111 patent does not cover Adobe Reader.

286.   Upon information and belief, Defendant has no reasonable basis to believe that claim 14 of the '111 patent covers Adobe Reader.

287.   Because no independent claim of the '111 patent covers Adobe Reader, the '111 patent does not cover Adobe Reader. Moreover, Defendant lacks any reasonable basis to believe that it does.

288.   U.S. Patent No. 6,754,382 ("the '382 patent"), entitled "IDENTIFYING COLOR SPACES FROM COLOR SPECIFICATIONS," issued on Jun. 22, 2004. (See http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=6,754,382.PN.&OS=PN/6,754,382&RS=

PN/6,754,382.)

289.   The '382 patent contains four independent claims.

290.   The first independent claim of the '382 patent reads as follows:

> 1. A method for identifying a color space from a color specification, comprising: receiving a color, specification, the color specification defining a mapping from an unknown color space to a well known color space; selecting a test color space, wherein the test color space is a color space used to identify the unknown color space; using the color specification to map the color components of a set of test colors from a representation in the test color space to a representation in the well known color space; using the color components of the set of test colors as represented in the well known color space to determine whether the color specification has correctly mapped the set of test colors from the representation in the test color space to the representation in the well known color space; and identifying the unknown color space as the test color space when the color specification has correctly mapped the set of test colors from the representation in the test color space to the representation in the well known color space.

291.   Upon information and belief, claim 1 of the '382 patent does not cover Adobe Reader.

292.   Upon information and belief, Defendant has no reasonable basis to believe that claim 1 of the '382 patent covers Adobe Reader.

293.   The second independent claim of the '382 patent reads as follows:

> 6. A method for verifying an association

between a stated color space and a color
specification, the method comprising: receiving a
stated color space; receiving a color specification,
the color specification tentatively defining a
mapping from the stated color space to a well
known color space; using the color specification to
map the color components of a set of test colors
from a representation in the stated color space to a
representation in the well known color space;
using the color components of the set of test colors
as represented in the well known color space to
determine whether the color specification has
correctly mapped the test colors from the
representation in the stated color space to the
representation in the well known color space; and
verifying the association between the stated color
space and the color specification when the color
specification has correctly mapped the set of test
colors from the representation in the stated color
space to the representation in the well known color
space.

294.   Upon information and belief, claim 6 of the '382 patent does not

cover Adobe Reader.

295.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 6 of the '382 patent covers Adobe Reader.

296.   The third independent claim of the '382 patent reads as follows:

11. A computer program product, stored on a
machine-readable medium, comprising instructions
operable to cause a programmable processor to:
receive a color specification, the color
specification defining a mapping from an unknown
color space to a well known color space; select a

- 83 -

test color space, wherein the test color space is a color space used to identify the unknown color space; use the color specification to map the color components of a set of test colors from a representation in the test color space to a representation in the well known color space; use the color components of the set of test colors as represented in the well known color space to determine whether the color specification has correctly mapped the set of test colors from the representation in the test color space to the representation in the well known color space; and identify the unknown color space as the test color space when the color specification has correctly mapped the set of test colors from the representation in the test color space to the representation in the well known color space.

297.   Upon information and belief, claim 11 of the '382 patent does not cover Adobe Reader.

298.   Upon information and belief, Defendant has no reasonable basis to believe that claim 11 of the '382 patent covers Adobe Reader.

299.   The final independent claim of the '382 patent reads as follows:

16. A computer program product, stored on a machine-readable medium, comprising instructions operable to cause a programmable processor to: receive a stated color space; receive a color specification, the color specification tentatively defining a mapping from the stated color space to a well known color space; use the color specification to map the color components of a set of test colors from a representation in the stated color space to a representation in the well known color space; use

the color components of the set of test colors as represented in the well known color space to determine whether the color specification has correctly mapped the test colors from the representation in the stated color space to the representation in the well known color space; and verify the association between the stated color space and the color specification when the color specification has correctly mapped the set of test colors from the representation in the stated color space to the representation in the well known color space.

300.   Upon information and belief, claim 16 of the '382 patent does not cover Adobe Reader.

301.   Upon information and belief, Defendant has no reasonable basis to believe that claim 16 of the '382 patent covers Adobe Reader.

302.   Because no independent claim of the '382 patent covers Adobe Reader, the '382 patent does not cover Adobe Reader. Moreover, Defendant lacks any reasonable basis to believe that it does.

303.   U.S. Patent No. 6,771,816 ("the '816 patent"), entitled "GENERATING A TEXT MASK FOR REPRESENTING TEXT PIXELS," issued on Aug. 3, 2004. (See http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO%2Fsrchnum.htm&r=1&f=G&l=50&s1=6,771,816.PN.&OS=PN/6,771,816&RS=PN/6,771,816.)

304.   The '816 patent contains two independent claims

305.   The first independent claim of the '816 patent reads as follows:

> 1. A method of processing an image
> including a non-coded lexical unit and a coded
> lexical unit, the non-coded lexical unit
> representing a unit of unrecognized text, the coded
> lexical unit representing a unit of recognized text,
> the method comprising: generating a text mask
> having pixels distinguishing between a plurality of
> text pixels and a plurality of local background
> pixels associated with the non-coded lexical unit in
> the image; generating a global background on
> which the non-coded lexical unit and the coded
> lexical unit is to be displayed, wherein one or more
> of the plurality of local background pixels
> associated with the non-coded lexical unit has a
> color mismatch relative to the global background;
> rendering the coded lexical unit on the global
> background according to a character code
> representing the coded lexical unit, the coded
> lexical unit having a pre-determined foreground
> color; and using the text mask to represent the non-
> coded lexical unit on the global background
> without displaying the one or more of the plurality
> of local background pixels associated with the
> non-coded lexical unit that has a color mismatch
> relative to the global background, including
> masking off the pixels of the text mask
> corresponding to the plurality of local background
> pixels and painting a text character through the
> pixels of the text mask corresponding to the
> plurality of text pixels, the text character being
> painted on the global background with the pre-
> determined foreground color.

306.   Upon information and belief, claim 1 of the '816 patent does not

cover Adobe Reader.

307.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 1 of the '816 patent covers Adobe Reader.

308.   The second independent claim of the '816 patent reads as follows:

> 13. A computer program product, tangibly
> stored on a computer-readable medium, for
> processing an image including a non-coded lexical
> unit and a coded lexical unit, the non-coded lexical
> unit representing a unit of unrecognized text, the
> coded lexical unit representing a unit of recognized
> text, the product comprising instructions operable
> to cause a programmable processor to: generate a
> text mask having pixels distinguishing between a
> plurality of text pixels and a plurality of local
> background pixels associated with the noncoded
> lexical unit in the image; generate a global
> background on which the non-coded lexical unit
> and the coded lexical unit is to be displayed,
> wherein one or more of the plurality of local
> background pixels associated with the non-coded
> lexical unit has a color mismatch relative to the
> global background; render the coded lexical unit
> on the global background according to a character
> code representing the coded lexical unit, the coded
> lexical unit having a pre-determined foreground
> color; and use the text mask to represent the non-
> coded lexical unit on the global background
> without displaying the one or more of the plurality
> of local background pixels associated with the
> non-coded lexical unit that has a color mismatch
> relative to the global background, including
> instructions to mask off the pixels of the text mask
> corresponding to the plurality of local background
> pixels and paint a text character through the pixels

of the text mask corresponding to the plurality of text pixels, the text character being painted on the global background with the pre-determined foreground color.

309.   Upon information and belief, claim 13 of the '816 patent does not cover Adobe Reader.

310.   Upon information and belief, Defendant has no reasonable basis to believe that claim 13 of the '816 patent covers Adobe Reader.

311.   Because no independent claim of the '816 patent covers Adobe Reader, the '816 patent does not cover Adobe Reader. Moreover, Defendant lacks any reasonable basis to believe that it does.

312.   U.S. Patent No. 6,842,786 ("the '786 patent"), entitled "METHOD AND APPARATUS FOR REMOTE COMMUNICATION OF DATA ASSOCIATED WITH DYNAMICALLY GENERATED TYPE OBJECT AT RUNTIME TO DESCRIBE THE DATA TYPE," issued on Jan. 11, 2005. (See http://patft.uspto.gov/netacgi/nph-Parser? Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO %2Fsrchnum.htm&r=1&f=G&l=50&s1=6,842,786.PN.&OS=PN/6,842,786&RS= PN/6,842,786.)

313.   The '786 patent contains four independent claims.

314.   The first independent claim of the '786 patent reads as follows:

1. A computer-implemented data processing method, comprising: running a first process in a first address space and a second process in a second address space, the first process including a request to send to the second process data having a data type; calling at runtime a type creation function to create a first type object describing the data type, the first type object having a set of associated functions for processing data, the set of associated functions including a marshalling function for encoding data and an unmarshalling function for decoding data; executing the marshalling function in the first process to generate encoded data, the marshalling function executed in the first process taking as input the data and the first type object; communicating the encoded data to the second process; and executing the unmarshalling function on the encoded data to decode the encoded data in the second process.

315. Upon information and belief, claim 1 of the '786 patent does not cover Adobe Reader.

316. Upon information and belief, Defendant has no reasonable basis to believe that claim 1 of the '786 patent covers Adobe Reader.

317. The second independent claim of the '786 patent reads as follows:

14. A computer-implemented data processing method, comprising: running a first process in a first format in a first address space and a second process in a second format in a second address space, the first process including a request to send to the second process data having a data type; calling at runtime in the first process a type creation function to create a first instance of a type

object describing the data type, the type object
having a set of associated functions for processing
data, the set of associated functions including a
marshalling function for encoding data and an
unmarshalling function for decoding data; calling
at runtime in the second process the type cordon
function to create a second instance of the type
object describing the data type; executing in the
first process the marshalling function, taking as
inputs the data and the first instance of the type
object to generate encoded data in a format
independent of the first and second formats;
communicating the encoded data from the first
process to the second process; and executing in the
second process the unmarshalling function to
decode the encoded data.

318.   Upon information and belief, claim 14 of the '786 patent does not

cover Adobe Reader.

319.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 14 of the '786 patent covers Adobe Reader.

320.   The third independent claim of the '786 patent reads as follows:

15. A computer program product, tangibly
stored on a computer-readable medium, for
processing data, comprising instructions operable
to cause a programmable processor to: run a first
process in a first address space and a second
process in a second address space, the first process
including a request to send to the second process
data having a data type; call at runtime a type
creation function to create a first type object
describing the data type, the first type object
having a set of associated functions for processing

data, the set of associated functions including a marshalling fiction for encoding data and an unmarshalling function for decoding data; execute the marshalling function in the first process to generate encoded data, the marshalling function executed in the first process taking as input the data and the first type object; communicate the encoded data to the second process; and execute the unmarshalling function on the encoded data to decode the encoded data in the second process.

321.   Upon information and belief, claim 15 of the '786 patent does not cover Adobe Reader.

322.   Upon information and belief, Defendant has no reasonable basis to believe that claim 15 of the '786 patent covers Adobe Reader.

323.   The final independent claim of the '786 patent reads as follows:

28. A computer program product, tangibly stored on a computer-readable medium, for processing data, comprising instructions operable to cause a programmable processor to: run a first process in a first format in a first address space and a second process in a second format in a second address space, the first process including a request to send to the second process data having a data type; call at runtime in the first process a type creation function to create a first instance of a type object describing the data type, the type object having a set of associated functions for processing data, the set of associated functions including a marshalling function for encoding data and an unmarshalling function for decoding data; call at runtime in the second process the type creation function to create a second instance of the type

object describing the data type; execute in the first process the marshalling function, the marshalling function taking as inputs the data and the first instance of the type object, to generate encoded data in a format independent of the first and second formats; communicate the encoded data from the first process to the second process; and execute in the second process the unmarshalling function to decode the encoded data.

324.   Upon information and belief, claim 28 of the '786 patent does not cover Adobe Reader.

325.   Upon information and belief, Defendant has no reasonable basis to believe that claim 28 of the '786 patent covers Adobe Reader.

326.   Because no independent claim of the '786 patent covers Adobe Reader, the '786 patent does not cover Adobe Reader. Moreover, Defendant lacks any reasonable basis to believe that it does.

327.   U.S. Patent No. 6,857,105 ("the '105 patent"), entitled "METHOD AND APPARATUS FOR EXPANDING AND CONTRACTING GRAPHICAL FUNCTION DISPLAYS," issued on Feb. 15, 2005. (See http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO%2Fsrchnum.htm&r=1&f=G&l=50&s1=6,857,105.PN.&OS=PN/6,857,105&RS=PN/6,857,105.)

328.   The '105 patent contains six independent claims.

329.   The first independent claim of the '105 patent reads as follows:

>   1. A computer-implemented graphical user interface method, comprising: providing a toolbar graphically representing one or more tools including one or more groups of tools, wherein a tool is represented by an icon and wherein a group of tools is represented by one or more icons, the toolbar including a first icon representing a first group of tools; and in response to a first user input, generating an expanded toolbar by expanding the toolbar to replace the first icon with a plurality of icons representing tools in the first group of tools; wherein the toolbar includes other icons and wherein generating an expanded toolbar to replace the first icon with a plurality of icons representing tools in the first group of tools comprises: removing the first icon from a position on the toolbar relative to the other icons included in the toolbar and including the plurality of icons in the same position on the toolbar relative to the other icons included in the toolbar.

330.   Upon information and belief, claim 1 of the '105 patent does not cover Adobe Reader.

331.   Upon information and belief, Defendant has no reasonable basis to believe that claim 1 of the '105 patent covers Adobe Reader.

332.   The second independent claim of the '105 patent reads as follows:

>   14. A computer-implemented graphical user interface method, comprising: providing a toolbar graphically representing one or more tools

including one or more groups of tool, wherein a tool is represented by an icon and wherein a group of tools is represented by one or more icons, the toolbar including a plurality of icons representing a plurality of tools in a first group of tools; and in response to a user input, generating a contracted toolbar by contracting the toolbar to replace the plurality of icons with one or more icons, but less than the plurality of icons, wherein the toolbar includes other icons and wherein contracting the toolbar to replace the plurality of icons with one or more icons, but less than the plurality of icons comprises: removing the plurality of icons from a position on the toolbar relative to the other icons included in the toolbar and including the one or more icons in the same position on the toolbar relative to the other icons included in the toolbar.

333.   Upon information and belief, claim 15 of the '105 patent does not

cover Adobe Reader.

334.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 15 of the '105 patent covers Adobe Reader.

335.   The third independent claim of the '105 patent reads as follows:

18. A computer-implemented graphical user interface method, comprising: providing a toolbar graphically representing one or more tools including one or more groups of tools, wherein a tool is represented by an icon and wherein a group of tools is represented by one or more icons, the toolbar including a first icon representing a first tool in a first group of tools and representing the first group of tools; in response to a first user input selecting the first icon, displaying a secondary list

including one or more icons representing the tools in the first group of tools; in response to a second user input, generating an expanded toolbar by expanding the toolbar to replace the first icon with a plurality of icons representing tools in the first group of tools and removing the secondary list; and in response to a third user input, contracting the expanded toolbar to generate a contracted toolbar, the contracted toolbar including one or more icons, but less than the plurality of icons, representing one or more tools in the first group of tools and representing the first group of tools.

336.   Upon information and belief, claim 18 of the '105 patent does not read on Adobe Reader.

337.   Upon information and belief, Defendant has no reasonable basis to believe that claim 18 of the '105 patent covers Adobe Reader.

338.   The fourth independent claim of the '105 patent reads as follows:

21. A computer program product, tangibly stored on a machine-readable medium, comprising instructions operable to cause a programmable processor to; provide a toolbar graphically representing one or more tools including one or more groups of tools, wherein a tool is represented by an icon and wherein a group of tools is represented by one or more icons, the toolbar including a first icon representing a first tool in a first group of tools and representing the first group of tools; and in response to a first user input, generate an expanded toolbar by expanding the toolbar to replace the first icon with a plurality of icons representing tools in the first group of tools; wherein the toolbar includes other icons and

wherein instructions operable to generate an expanded toolbar to replace the first icon with a plurality of icons representing tools in the first group of tools comprise instructions operable to: remove the first icon from a position on the toolbar relative to the other icons included in the toolbar and include the plurality of icons in the same position on the toolbar relative to the other icons included in the toolbar.

339.   Upon information and belief, claim 21 of the '105 patent does not

cover Adobe Reader.

340.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 21 of the '105 patent covers Adobe Reader.

341.   The fifth independent claim of the '105 patent reads as follows:

34. A computer program product, tangibly stored on a machine-readable medium, comprising instructions operable to cause a programmable processor to: provide a toolbar graphically representing one or more tools including one or more groups of tool, wherein a tool is represented by an icon and wherein a group of tools is represented by one or more icons, the toolbar including a plurality of icons representing a plurality of tools in a first group of tools; and in response to a user input, generate a contracted toolbar by contracting the toolbar to replace the plurality of icons with one or more icons, but less than the plurality icons, wherein the toolbar includes other icons and wherein contracting the toolbar to replace the plurality of icons with one or more icons, but less than the plurality of icons comprises: removing the plurality of icons from a

position on the toolbar relative to the other icons included in the toolbar and including the one or more icons in the same position on the toolbar relative to the other icons included in the toolbar.

342.   Upon information and belief, claim 34 of the '105 patent does not cover Adobe Reader.

343.   Upon information and belief, Defendant has no reasonable basis to believe that claim 34 of the '105 patent covers Adobe Reader.

344.   The final independent claim of the '105 patent reads as follows:

> 38. A computer program product, tangibly stored on a machine-readable medium, comprising instructions operable to cause a programmable processor to: provide a toolbar graphically representing one or more tools including one or more groups of tools, wherein a tool is represented by an icon and wherein a group of tools is represented by one or more icons, the toolbar including a first icon representing a first tool in a first group of tools and representing the first group of tools; in response to a first user input selecting the first icon, display a secondary list including one or more icons representing the tools in the first group of tools; in response to a second user input, generate an expanded toolbar by expanding the toolbar to replace the first icon with a plurality of icons representing tools in the first group of tools and remove the secondary list; and in response to a third user input, contract the expanded toolbar to generate a contracted toolbar, the contracted toolbar including one or more icons, but less than the plurality of icons, representing one or more tools in the first group of tools and representing the

first group of tools.

345.   Upon information and belief, claim 38 of the '105 patent does not

cover Adobe Reader.

346.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 38 of the '105 patent covers Adobe Reader.

347.   Because no independent claim of the '105 patent covers Adobe

Reader, the '105 patent does not cover Adobe Reader. Moreover, Defendant lacks

any reasonable basis to believe that it does.

348.   U.S. Patent No. 6,894,704 ("the '704 patent"), entitled "Processing

complex regions of illustration artwork," issued on May 17, 2005. (See

http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=6,894,704.PN.&OS=PN/6,894,704&RS=

PN/6,894,704.)

349.   The '704 patent contains two independent claims.

350.   The first independent claim of the '704 patent reads as follows:

> 1. A computer-based method of processing a
> computer graphics illustration that includes one or
> more pieces of artwork, the method comprising:
> mapping outlines of at least one of the pieces of
> artwork onto a grid of cells, a piece of artwork
> having one outline; determining the total number

of outlines of pieces of artwork that map to a cell
of the grid; identifying the cell as a complex region
based on the total number of outlines that map to
the cell; and identifying pieces of artwork to
include in an illustration flattening process based
on the identification of the complex region.

351.   Upon information and belief, claim 1 of the '704 patent does not

cover Adobe Reader.

352.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 1 of the '704 patent covers Adobe Reader.

353.   The second independent claim of the '704 patent reads as follows:

14. A computer program product, tangibly
stored on machine-readable medium, for
processing a computer graphics illustration having
pieces of artwork, the product comprising
instructions operable to cause a processor to: map
outlines of at least one of the pieces of artwork
onto a grid of cells, a piece of artwork having one
outline; determine the total number of outlines of
pieces of artwork that map to a cell of the grid;
identify the cell as a complex region based on the
total number of outlines that map to the cell; and
exclude, based on the identifying of the cell as a
complex region, pieces of artwork from an
illustration flattening process.

354.   Upon information and belief, claim 14 of the '704 patent does not

cover Adobe Reader.

355.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 14 of the '704 patent covers Adobe Reader.

356.   Because no independent claim of the '704 patent covers Adobe

Reader, the '704 patent does not cover Adobe Reader. Moreover, Defendant lacks

any reasonable basis to believe that it does.

357.   U.S. Patent No. 6,915,484 ("the '484 patent"), entitled "TEXT

REFLOW IN A STRUCTURED DOCUMENT," issued on Jul. 5, 2005. (See

http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=6,915,484.PN.&OS=PN/6,915,484&RS=

PN/6,915,484.)

358.   The '484 patent contains two independent claims.

359.   The first independent claim of the '484 patent reads as follows:

> 1. A computer program product, tangibly
> stored on a computer-readable medium, for
> reflowing a page, the product comprising
> instructions operable to cause a programmable
> processor to: receive a page represented in a page
> description language, the page including a plurality
> of columns of text positioned side-by-side on the
> page, each of the plurality of columns having a
> width and a length, the page further including a
> graphical element having a vertical position on the
> page, adjacent to a vertical side of one of and
> outside each of the columns; for each of the
> plurality of columns, associate an item of text in
> the column with the graphical element, the item

and the graphical element having a spatial
relationship in that the item has a vertical position
corresponding to the vertical position of the
graphical element; for at least one of the plurality
of columns, change a size of the text, the width of
the column, or both so that the relationship
between the size of the text and the width of the
column is changed, and then reflow text in the
plurality of columns so that the item of text in each
of the columns has a new vertical position as a
result of the reflow; for each of the items, calculate
a change in vertical position resulting from the
reflow; and reposition the graphical element on the
page according to the new vertical position of the
item of text having the greatest change in vertical
position.

360. Upon information and belief, claim 1 of the '484 patent does not

cover Adobe Reader.

361. Upon information and belief, Defendant has no reasonable basis to

believe that claim 1 of the '484 patent covers Adobe Reader.

362. The second independent claim of the '484 patent reads as follows:

10. A method for reflowing a page,
comprising: electronically receiving a page
represented in a page description language, the
page including a plurality of columns of text
positioned side-by-side on the page, each of the
plurality of columns having a width and a length,
the page further including a graphical element
having a vertical position on the page, adjacent to a
vertical side of one of and outside each of the
columns; for each of the plurality of columns,
associating an item of text in the column with the

graphical element, the item and the graphical element having a spatial relationship in that the item has a vertical position corresponding to the vertical position of the graphical element; for at least one of the plurality of the columns, changing a size of the text, the width of the column, or both so that the relationship between the size of the text and the width of the column is changed, and then reflowing text in the plurality of columns so that the item of text in each of the columns has a new vertical position as result of the reflow; for each of the, items calculating a change in vertical position resulting from the reflow; and repositioning the graphical element on the page according to the new vertical position of the item of text having the greatest change in vertical position.

363.   Upon information and belief, claim 10 of the '484 patent does not cover Adobe Reader.

364.   Upon information and belief, Defendant has no reasonable basis to believe that claim 10 of the '484 patent covers Adobe Reader.

365.   Because no independent claim of the '484 patent covers Adobe Reader, the '484 patent does not cover Adobe Reader. Moreover, Defendant lacks any reasonable basis to believe that it does.

366.   U.S. Patent No. 6,934,909 ("the '909 patent"), entitled "IDENTIFYING LOGICAL ELEMENTS BY MODIFYING A SOURCE DOCUMENT USING MARKER ATTRIBUTE VALUES," issued on Aug. 23, 2005. (See http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=6,934,909.PN.&OS=PN/6,934,909&RS=

PN/6,934,909.)

367.   The '909 patent contains two independent claims.

368.   The first independent claim of the '909 patent reads as follows:

> 1. A computer-implemented method for processing an electronic document, the method comprising: obtaining a first electronic document including a plurality of logical elements, each logical element having associated content with a visual appearance; modifying the first electronic document by associating a respective marker attribute value with a marker attribute of each of the logical elements in the first electronic document, each respective marker attribute value being a value of the marker attribute of the content of the respective logical element; generating a second electronic document by converting the modified first electronic document with the associated marker attribute values through a document conversion process that preserves the association of the marker attribute values and the content of the logical elements; using the marker attribute values of the content in the second electronic document to identify each of the plurality of logical elements in the second electronic document; generating a third electronic document before associating a marker attribute value with each of the plurality of logical elements in the first electronic document, by converting the first electronic document through the document conversion process; and using the marker attribute values in the second electronic document to

identify logical elements in the third electronic document.

369.   Upon information and belief, claim 1 of the '909 patent does not cover Adobe Reader.

370.   Upon information and belief, Defendant has no reasonable basis to believe that claim 1 of the '909 patent covers Adobe Reader.

371.   The second independent claim of the '909 patent reads as follows:

> 12. A computer program product, stored on a machine-readable medium, comprising instructions operable to cause a programmable processor to: obtain a first electronic document including a plurality of logical elements, each logical element having associated content with a visual appearance; modify the first electronic document by associating a respective market attribute value with a marker attribute of each of the logical elements in the first electronic document, each respective marker attribute value being a value of the marker attribute of the content of the respective logical element; generate a second electronic document by converting the modified first electronic document with the associated marker attribute values through a document conversion process that preserves the association of the marker attribute values and the content of the logical elements; use the marker attribute values of the content in the second electronic document to identify each of the plurality of logical elements in the second electronic document; generate a third electronic document before associating a marker attribute value with each of the plurality of logical elements in the first electronic document, by

converting the first electronic document through the document conversion process; and use the marker attribute values in the second electronic document to identify logical elements in the third electronic document.

372.   Upon information and belief, claim 12 of the '909 patent does not cover Adobe Reader.

373.   Upon information and belief, Defendant has no reasonable basis to believe that claim 12 of the '909 patent covers Adobe Reader.

374.   Because no independent claim of the '909 patent covers Adobe Reader, the '909 patent does not cover Adobe Reader. Moreover, Defendant lacks any reasonable basis to believe that it does.

375.   U.S. Patent No. 6,964,051 ("the '051 patent"), entitled "ENABLING AND DISABLING REMOTE FEATURES DYNAMICALLY IN SOFTWARE APPLICATIONS," issued on Nov. 8, 2005. (See

http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=6,964,051.PN.&OS=PN/6,964,051&RS=

PN/6,964,051.)

376.   The '051 patent contains three independent claims.

377.   The first independent claim of the '051 patent reads as follows:

1. A computer program product, tangibly stored on a computer readable medium, comprising: instructions operable to execute on a client computer to interact with one or more functionality servers remote from the client computer to obtain, and to maintain locally on the client computer, control information specifying the availability of each of a set of one or more remote features, each remote feature being a feature that requires, for proper operation in an application that invokes the remote feature, a resource external to the application; and instructions operable to execute as a shared component on a client computer for multiple distinct applications, the instructions of the shared component being operable to cause the shared component (i) to receive requests for one or more particular remote features from two or more applications, and (ii) to use the obtained control information to respond to each request with either (a) information indicating that the requested remote feature should be treated as unavailable, or (b) information indicating that the requested remote feature should be treated as available; whereby each distinct application operating with the shared component can determine whether or not remote features should be enabled in a user interface provided by the application without the application having to access a remote server.

378.   Upon information and belief, claim 1 of the '051 patent does not cover Adobe Reader.

379.   Upon information and belief, Defendant has no reasonable basis to believe that claim 1 of the '051 patent covers Adobe Reader.

380.   The second independent claim of the '051 patent reads as follows:

12. A system for dynamically enabling remote features in computer program applications, the system comprising: one or more functionality servers, each functionality server operable to communicate with a remote client computer, each functionality server maintaining control information concerning one or more remote features, each remote feature being a feature that requires, for proper operation in an application that runs on the client computer and that invokes the remote feature, a resource external to the application; a monitoring process running on the client computer, the monitoring process maintaining a local cache copy of the control information by periodic synchronization with one or more of the functionality servers; a feature-enabling process configured to execute in each of two or more distinct computer program applications, each application having user interface hooks for the one or more remote features, the feature-enabling process being operable to use the control information maintained in the cache copy to determine whether to treat a particular remote feature as being available before the application provides a user an option of invoking the particular remote feature through a corresponding user interface hook, whereby each application is operable to disable hooks for remote features determined by the feature-enabling process not be available.

381.   Upon information and belief, claim 12 of the '051 patent does not

cover Adobe Reader.

382.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 12 of the '051 patent covers Adobe Reader.

383.   The final independent claim of the '051 patent reads as follows:

21. Apparatus configured to communicate with one or more remote servers, the apparatus comprising: a client computer; a shared component installed on the client computer and accessible by one or more applications installed on the client computer, the shared component operable to: interact with the one or more functionality servers to obtain, and to maintain locally on the client computer, control information specifying the availability of each of a set of one or more remote features, each remote feature being a feature that requires, for proper operation in an application that invokes the remote feature, a resource external to the application; and operate as a shared component with multiple distinct applications, (i) to receive requests for one or more particular remote features from two or more applications, and (ii) to use the obtained control information to respond to each request with either (a) information indicating that the requested remote feature should be treated as unavailable, or (b) information indicating that the requested remote feature should be treated as available; whereby each distinct application operating with the shared component can determine whether or not remote features should be enabled in a user interface provided by the application without the application having to access a remote server.

384.   Upon information and belief, claim 21 of the '051 patent does not

cover Adobe Reader.

385.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 21 of the '051 patent covers Adobe Reader.

386.   Because no independent claim of the '051 patent covers Adobe Reader, the '051 patent does not cover Adobe Reader. Moreover, Defendant lacks any reasonable basis to believe that it does.

387.   U.S. Patent No. 7,071,941 ("the '941 patent"), entitled "Method for calculating CJK emboxes in fonts," issued on Jul. 4, 2006. (See http://patft.uspto.gov/netacgi/nph-Parser? Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO %2Fsrchnum.htm&r=1&f=G&l=50&s1=7,071,941.PN.&OS=PN/7,071,941&RS= PN/7,071,941.)

388.   The '941 patent contains three independent claims.

389.   The first independent claim of the '941 patent reads as follows:

> 1. A method for typesetting a text line, comprising: determining dimensions of each of a plurality of characters to be typeset on a text line; associating a distinct embox with each of the plurality of characters, each embox having an outer frame demarcated by the dimensions of the associated character of the plurality of characters; associating a reference character with each of the plurality of characters, each associated reference character being one of one or more preselected reference characters, each reference character of the one or more preselected reference characters having the same font and dimensions as the associated character of the plurality of characters;

using the associated reference character to determine a coordination point for each distinct embox and thereby determine a coordination point for each of the plurality of characters; and aligning each distinct embox with a coordination line using the determined coordination point for each of the plurality of characters to typeset the plurality of characters on the text line.

390.   Upon information and belief, claim 1 of the '941 patent does not cover Adobe Reader.

391.   Upon information and belief, Defendant has no reasonable basis to believe that claim 1 of the '941 patent covers Adobe Reader.

392.   The second independent claim of the '941 patent reads as follows:

13. A computer program product, stored on a machine-readable medium, comprising instructions operable to cause a programmable processor to: select a coordination line for the text line to coordinate a plurality of characters for typesetting the text line, determine the dimensions of each of the plurality of characters to be typeset on the text line, associate a distinct embox with each of the plurality of characters, each embox having an outer frame demarcated by the dimensions of the associated character of the plurality of characters; associate a reference character with each of the plurality of characters, each associated reference character being one of one or more preselected reference characters, each reference character of the one or more preselected reference characters having the same font and dimensions as the associated character of the plurality of characters; use the associated reference character to determine

a coordination point for each distinct embox and thereby determine a coordination point for each of the plurality of characters; and align each distinct embox with the coordination line using the determined coordination point for each of the plurality of characters to typeset the plurality of characters on the text line.

393.   Upon information and belief, claim 13 of the '941 patent does not

cover Adobe Reader.

394.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 13 of the '941 patent covers Adobe Reader.

395.   The final independent claim of the '941 patent reads as follows:

25. A desktop publishing system for controlling typesetting of a text line, comprising: a desktop publishing processing control device provided with a font file and with typesetting control means, the font file containing character font information for performing typesetting; a display device for displaying data that is being typeset; and input means for receiving user input; said typesetting control means having a text line typesetting control means adapted to arranging a plurality of characters to be coordinated with a coordination line of a text line by determining dimensions of each of the plurality of characters to be typeset on the text line, associating a distinct embox with each of the plurality of characters, each embox having an outer frame demarcated by the dimensions of the associated character of the plurality of characters, associating with each of the plurality of characters a reference character, each associated reference character being one of one of

more preselected reference characters, each
reference character of the one or more preselected
reference characters having the same font and
dimensions as the associated character of the
plurality of characters, using the associated
reference character to determine a coordination
point for each distinct embox and thereby
determine a coordination point for each of the
plurality of characters, and performing line
typesetting processing by aligning each distinct
embox with the coordination line using to
determined coordination point for each of the
plurality of characters.

396.   Upon information and belief, claim 25 of the '941 patent does not

cover Adobe Reader.

397.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 25 of the '941 patent covers Adobe Reader.

398.   Because no independent claim of the '941 patent covers Adobe

Reader, the '941 patent does not cover Adobe Reader. Moreover, Defendant lacks

any reasonable basis to believe that it does.

399.   U.S. Patent No. 7,089,248 ("the '248 patent"), entitled "Group file

delivery including user-defined metadata," issued on Aug. 8, 2006. (See

http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=7,089,248.PN.&OS=PN/7,089,248&RS=

PN/7,089,248.)

400.   The '248 patent contains three independent claims.

401.   The first independent claim of the '248 patent reads as follows:

> 1. A computer-implemented method, comprising: receiving user input specifying files to be embedded in an electronic document and metadata to be associated with the specified files in the electronic document; embedding the specified files in the electronic document in response to the input; receiving user input specifying behavior to be associated with one or more of the embedded files; and creating a manifest in the electronic document, the manifest including representations of each of the embedded files and the specified metadata, each representation of the metadata being associated in the manifest with representations of one or more corresponding files of the embedded files, the manifest including presentation information defining how the representations will be presented to a user, the manifest including one or more interactive controls, the interactive controls being selectable by a user to perform one or more actions defined by the behavior specified for the corresponding embedded files.

402.   Upon information and belief, claim 1 of the '248 patent does not cover Adobe Reader.

403.   Upon information and belief, Defendant has no reasonable basis to believe that claim 1 of the '248 patent covers Adobe Reader.

404.   The second independent claim of the '248 patent reads as follows:

16. A computer-implemented method, comprising: receiving user input specifying files to be embedded in an electronic document and metadata to be associated with the specified files in the electronic document; embedding the specified files in the electronic document in response to the input; receiving user input specifying behavior to be associated with one or more of the embedded files; providing a manifest template, the manifest template defining a default layout for representations of one or more of the embedded files and the metadata; and creating a manifest in the electronic document according to the manifest template, the manifest including representations of each of the embedded files and the specified metadata, each representation of the metadata being associated in the manifest with representations of one or more corresponding files of the embedded files, the manifest including presentation information defining how the representations will be presented to a user, the manifest including one or more interactive controls, the interactive controls being selectable by a user to perform one or more actions defined by the behavior specified for the corresponding embedded files.

405.   Upon information and belief, claim 16 of the '248 patent does not cover Adobe Reader.

406.   Upon information and belief, Defendant has no reasonable basis to believe that claim 16 of the '248 patent covers Adobe Reader.

407.   The final independent claim of the '248 patent reads as follows:

22. A computer program product, tangibly

stored on a computer-readable medium,
comprising instructions operable to cause a
computer system to: receive user input specifying
files to be embedded in an electronic document
and metadata to be associated with the specified
files in the electronic document; embed the
specified files in the electronic document in
response to the input; and receive user input
specifying behavior to be associated with one or
more of the embedded files; and create a manifest
in the electronic document, the manifest including
representations of each of the embedded files and
the specified metadata, each representation of the
metadata being associated in the manifest with
representations of one or more corresponding files
of the embedded files, the manifest including
presentation information defining how the
representations will be presented to a user, the
manifest including one or interactive controls, the
interactive controls being selectable by a user to
perform one or more actions defined by the
behavior specified for the corresponding
embedded files.

408.   Upon information and belief, claim 22 of the '248 patent does not

cover Adobe Reader.

409.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 22 of the '248 patent covers Adobe Reader.

410.   Because no independent claim of the '248 patent covers Adobe

Reader, the '248 patent does not cover Adobe Reader. Moreover, Defendant lacks

any reasonable basis to believe that it does.

411.   U.S. Patent No. 7,246,748 ("the '748 patent"), entitled "Enabling the

use of machine-readable codes," issued on Jul. 24, 2007. (See

http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=7,246,748.PN.&OS=PN/7,246,748&RS=

PN/7,246,748.)

412.   The '748 patent contains eleven independent claims.

413.   The first independent claim of the '748 patent reads as follows:

> 1. A computer program product, stored on a
> machine-readable medium, the computer program
> product comprising instructions operable to cause
> a data processing apparatus to: receive code data
> generated by reading a machine-readable code, the
> code data comprising a payload; detect the
> presence of formatting information (FI) in the code
> data, the FI indicating whether the payload is in a
> first format or a second, distinct format, where the
> first format is an open format and the second
> format is a closed format; if the FI is not detected,
> or if the FI is detected and the FI indicates that the
> payload is in the first format, use a first decode
> procedure to decode the payload; and if the FI is
> detected and the FI indicates that the payload is in
> the second format, use a second, distinct decode
> procedure to decode the payload.

414.   Upon information and belief, claim 1 of the '748 patent does not

cover Adobe Reader.

415. Upon information and belief, Defendant has no reasonable basis to believe that claim 1 of the '748 patent covers Adobe Reader.

416. The second independent claim of the '748 patent reads as follows:

> 16. A computer-implemented method comprising: receiving code data generated by reading a machine-readable code, the code data comprising a payload; detecting the presence of formatting information (FI) in the code data, the FI indicating whether the payload is in a first format or a second, distinct format, where the first format is an open format and the second format is a closed format; if the FI is not detected, or if the FI is detected and the FI indicates that the payload is in the first format, using a first decode procedure to decode the payload; and if the FI is detected and the FI indicates that the payload is in the second format, using a second, distinct decode procedure to decode the payload.

417. Upon information and belief, claim 16 of the '748 patent does not cover Adobe Reader.

418. Upon information and belief, Defendant has no reasonable basis to believe that claim 16 of the '748 patent covers Adobe Reader.

419. The third independent claim of the '748 patent reads as follows:

> 17. A system comprising: means for receiving code data generated by reading a machine-readable code, the code data comprising a payload; means for detecting the presence of formatting information (FI) in the code data, the FI indicating whether the payload is in a first format

or a second, distinct format, where the first format
is an open format and the second format is a closed
format; means for using a first decode procedure to
decode the payload if the FI is not detected or if
the FI is detected and the FI indicates that the
payload is in the first format; and means for using
a second, distinct decode procedure to decode the
payload if the FI is detected and the FI indicates
that the payload is in the second format.

420.   Upon information and belief, claim 17 of the '748 patent does not

cover Adobe Reader.

421.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 17 of the '748 patent covers Adobe Reader.

422.   The fourth independent claim of the '748 patent reads as follows:

18. A computer program product, stored on a
machine-readable medium, the computer program
product comprising instructions operable to cause
a data processing apparatus to: determine whether
a machine-readable code is authorized to be
produced in a first format, wherein the machine-
readable code is to be included in a final
representation of an electronic document, the
electronic document comprising one or more fields
for receiving user data, and wherein the machine-
readable code is to encode the user data so that the
user data can be reproduced by decoding the
machine-readable code; if the machine-readable
code is authorized to be produced in the first
format, set an encoding instruction to indicate to a
program operable to receive the user data and to
generate the final representation of the electronic
document that the machine-readable code is to be

produced in the first format; if the machine-readable code is not authorized to be produced in the first format, set the encoding instruction to indicate that the machine-readable code is to be produced in a second, distinct format; and associate the encoding instruction with the electronic document.

423.   Upon information and belief, claim 18 of the '748 patent does not cover Adobe Reader.

424.   Upon information and belief, Defendant has no reasonable basis to believe that claim 18 of the '748 patent covers Adobe Reader.

425.   The fifth independent claim of the '748 patent reads as follows:

34. A computer-implemented method comprising: determining whether a machine-readable code is authorized to be produced in a first format, wherein the machine-readable code is to be included in a final representation of an electronic document, the electronic document comprising one or more fields for receiving user data, and wherein the machine-readable code is to encode the user data so that the user data can be reproduced by decoding the machine-readable code; if the machine-readable code is authorized to be produced in the first format, setting an encoding instruction to indicate to a program operable to receive the user data and to generate the final representation of the electronic document that the machine-readable code is to be produced in the first format; if the machine-readable code is not authorized to be produced in the first format setting the encoding instruction to indicate that the machine-readable code is to be produced in a

second, distinct format; and associating the encoding instruction with the electronic document.

426. Upon information and belief, claim 34 of the '748 patent does not cover Adobe Reader.

427. Upon information and belief, Defendant has no reasonable basis to believe that claim 34 of the '748 patent covers Adobe Reader.

428. The sixth independent claim of the '748 patent reads as follows:

35. A system comprising: means for determining whether a machine-readable code is authorized to be produced in a first format, wherein the machine-readable code is to be included in a final representation of an electronic document, the electronic document comprising one or more fields for receiving user data, and wherein the machine-readable code is to encode the user data so that the user data can be reproduced by decoding the machine-readable code; means for setting an encoding instruction to indicate to a program operable to receive the user data and to generate the final representation of the electronic document that the machine-readable code is to be produced in the first format, if the machine-readable code is authorized to be produced in the first format; means for setting the encoding instruction to indicate that the machine-readable code is to be produced in a second, distinct format, if the machine-readable code is not authorized to be produced in the first format; and means for associating the encoding instruction with the electronic document.

429. Upon information and belief, claim 35 of the '748 patent does not

cover Adobe Reader.

430.   Upon information and belief, Defendant has no reasonable basis to believe that claim 35 of the '748 patent covers Adobe Reader.

431.   The seventh independent claim of the '748 patent reads as follows:

> 36. A computer program product, stored on a machine-readable medium, the computer program product comprising instructions operable to cause a data processing apparatus to: receive an electronic document, the electronic document comprising one or more fields for receiving user data to be encoded into a machine-readable code; receive user input comprising the user data to be encoded into the machine-readable code; inspect an encoding instruction associated with the electronic document to determine whether to produce the machine-readable code in a first format or a second, distinct format, where the first format is an open format and the second format is a closed format; and produce the machine-readable code in the first format or the second format depending on the encoding instruction, the machine-readable code being included in a final representation of the electronic document and encoding the user data so that the user data can be reproduced by decoding the machine-readable code.

432.   Upon information and belief, claim 36 of the '748 patent does not cover Adobe Reader.

433.   Upon information and belief, Defendant has no reasonable basis to believe that claim 36 of the '748 patent covers Adobe Reader.

434.   The eighth independent claim of the '748 patent reads as follows:

> 56. A computer-implemented method comprising: receiving an electronic document, the electronic document comprising one or more fields for receiving user data to be encoded into a machine-readable code; receiving user input comprising the user data to be encoded into the machine-readable code; inspecting an encoding instruction associated with the electronic document to determine whether to produce the machine-readable code in a first format or a second, distinct format, where the first format is an open format and the second format is a closed format; and producing the machine-readable code in the first format or the second format depending on the encoding instruction, the machine-readable code being included in a final representation of the electronic document and encoding the user data so that the user data can be reproduced by decoding the machine-readable code.

435.   Upon information and belief, claim 56 of the '748 patent does not

cover Adobe Reader.

436.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 56 of the '748 patent covers Adobe Reader.

437.   The ninth independent claim of the '748 patent reads as follows:

> 57. A system comprising: means for receiving an electronic document, the electronic document comprising one or more fields for receiving user data to be encoded into a machine-readable code; means for receiving user input comprising the user data to be encoded into the

machine-readable code; means for inspecting an encoding instruction associated with the electronic document to determine whether to produce the machine-readable code in a first format or a second, distinct format, where the first format is an open format and the second format is a closed format; and means for producing the machine-readable code in the first format or the second format depending on the encoding instruction, the machine-readable code being included in a final representation of the electronic document and encoding the user data so that the user data can be reproduced by decoding the machine-readable code.

438.   Upon information and belief, claim 57 of the '748 patent does not cover Adobe Reader.

439.   Upon information and belief, Defendant has no reasonable basis to believe that claim 57 of the '748 patent covers Adobe Reader.

440.   The tenth independent claim of the '748 patent reads as follows:

58. A computer-implemented method comprising: generating an electronic document, the electronic document comprising one or more fields for receiving user data; determining whether a machine-readable code to be included in a final representation of the electronic document is authorized to be produced in a first format or a second, distinct format; setting an encoding instruction to indicate whether the machine-readable code is authorized to be produced in the first format or the second format; associating the encoding instruction with the electronic document; receiving user input comprising the user data;

inspecting the encoding instruction associated with the electronic document to determine whether to produce the machine-readable code in the first format or the second format; producing the machine-readable code by encoding the user data in the first format or the second format depending on the encoding instruction, the machine-readable code being included in the final representation of the electronic document; reading the machine-readable code in the final representation of the electronic document to generate code data; determining whether the code data is in the first format or the second format; and reproducing the user data by decoding the code data according to a first decode procedure if the code data is in the first format, or according to a second, distinct decode procedure if the code data is in the second format, wherein decoding the code data according to the second decode procedure comprises determining whether use of the second decode procedure is authorized.

441.   Upon information and belief, claim 58 of the '748 patent does not cover Adobe Reader.

442.   Upon information and belief, Defendant has no reasonable basis to believe that claim 58 of the '748 patent covers Adobe Reader.

443.   The final independent claim of the '748 patent reads as follows:

59. A system comprising: means for generating an electronic document, the electronic document comprising one or more fields for receiving user data; means for determining whether a machine-readable code to be included in a final representation of the electronic document is

authorized to be produced in a first format or a second, distinct format; means for setting an encoding instruction to indicate whether the machine-readable code is authorized to be produced in the first format or the second format; means for associating the encoding instruction with the electronic document; means for receiving user input comprising the user data; means for inspecting the encoding instruction associated with the electronic document to determine whether to produce the machine-readable code in the first format or the second format; means for producing the machine-readable code by encoding the user data in the first format or the second format depending on the encoding instruction, the machine-readable code being included in the final representation of the electronic document; means for reading the machine-readable code in the final representation of the electronic document to generate code data; means for determining whether the code data is in the first format or the second format; and means for reproducing the user data by decoding the code data according to a first decode procedure if the code data is in the first format, or according to a second, distinct decode procedure if the code data is in the second format, wherein decoding the code data according to the second decode procedure comprises determining whether use of the second decode procedure is authorized.

444.   Upon information and belief, claim 59 of the '748 patent does not cover Adobe Reader.

445.   Upon information and belief, Defendant has no reasonable basis to believe that claim 59 of the '748 patent covers Adobe Reader.

446.   Because no independent claim of the '748 patent covers Adobe Reader, the '748 patent does not cover Adobe Reader. Moreover, Defendant lacks any reasonable basis to believe that it does.

447.   U.S. Patent No. 7,305,617 ("the '617 patent"), entitled "Method for aligning text to baseline grids and to CJK character grids," issued on Dec. 4, 2007. (See http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=7,305,617.PN.&OS=PN/7,305,617&RS=

PN/7,305,617.)

448.   The '617 patent contains four independent claims.

449.   The first independent claim of the '617 patent reads as follows:

> 1. A computer program product, stored on a computer-readable recording medium, comprising instructions operable to cause a programmable processor to: determine the height of text consisting of a plurality of characters to be arranged within a current line in a grid displayed on a display device, the grid comprising a plurality of grid lines, each grid line including a plurality of cells for arranging characters within the grid line according to a particular coordination mode; when the height of the text is larger than a specified dimension for the grid, demarcate an arrangement region that includes the current line and at least one subsequent line in the grid, where the arrangement region defines a new line with respect to the grid for arranging the plurality of characters;

set a coordination line for the arrangement region according to a selected coordination mode; arrange the plurality of characters within the arrangement region while coordinating the plurality of characters with the coordination line; and displaying the arranged plurality of characters.

450.   Upon information and belief, claim 1 of the '617 patent does not cover Adobe Reader.

451.   Upon information and belief, Defendant has no reasonable basis to believe that claim 1 of the '617 patent covers Adobe Reader.

452.   The second independent claim of the '617 patent reads as follows:

> 10. A method for controlling forced grid line spacing, comprising: determining the height of text that includes a plurality of characters to be arranged within a current line in a grid displayed on a display device, the grid comprising a plurality of grid lines, each grid line including a plurality of cells for arranging characters within the grid line according to a particular coordination mode; when the height of the text is larger than a specified dimension for the grid, demarcating an arrangement region that includes the current line and at least one subsequent line in the grid, where the arrangement region defines a new line with respect to the grid for arranging the plurality of characters; setting a coordination line for the arrangement region according to a selected coordination mode; arranging the plurality of characters within the arrangement region while coordinating the plurality of characters with the coordination line; and displaying the arranged plurality of characters.

453.   Upon information and belief, claim 10 of the '617 patent does not cover Adobe Reader.

454.   Upon information and belief, Defendant has no reasonable basis to believe that claim 10 of the '617 patent covers Adobe Reader.

455.   The third independent claim of the '617 patent reads as follows:

> 17. A desktop publishing system for controlling forced grid line spacing, comprising: a desktop publishing processing control device provided with a font file, the font file storing character font information for performing typesetting, and with typesetting control means having a control means for forced grid line spacing; a display device displaying data being typeset; and input means for user input; the control means for forced grid line spacing being arranged to: determine whether a maximum dimension of a plurality of characters to be arranged according to a selected coordination mode within a current line of a grid displayed on the display device exceeds a specified dimension of the grid, the grid comprising a plurality of grid lines, each grid line including a plurality of cells for arranging characters within the grid line according to a particular coordination mode; and when the maximum dimension of the plurality of characters exceeds the specified dimension: select a plurality of grid lines including a current grid line and at least one subsequent grid line as an arrangement space, where the arrangement space defines a new line with respect to the grid for arranging the plurality of characters; and arrange the plurality of characters within the arrangement space, based on

the coordination mode.

456.   Upon information and belief, claim 17 of the '617 patent does not

cover Adobe Reader.

457.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 17 of the '617 patent covers Adobe Reader.

458.   The final independent claim of the '617 patent reads as follows:

> 19. A method for controlling forced grid line
> spacing, comprising: determining whether a
> maximum dimension of a plurality of characters to
> be arranged according to a selected coordination
> mode within a current line of a grid displayed on a
> display device exceeds a specified dimension of
> the grid, the grid comprising a plurality of grid
> lines, each grid line including a plurality of cells
> for arranging characters within the grid line
> according to a particular coordination mode; and
> when the maximum dimension exceeds a specified
> dimension of the grid: selecting a plurality of grid
> lines including a current grid line and at least one
> subsequent grid line as an arrangement space,
> where the arrangement space defines a new line
> with respect to the grid for arranging the plurality
> of characters; and arranging the plurality of
> characters within the arrangement space, based on
> the selected coordination mode.

459.   Upon information and belief, claim 19 of the '617 patent does not

cover Adobe Reader.

460.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 19 of the '617 patent covers Adobe Reader.

461. Because no independent claim of the '617 patent covers Adobe Reader, the '617 patent does not cover Adobe Reader. Moreover, Defendant lacks any reasonable basis to believe that it does.

462. U.S. Patent No. 7,310,769 ("the '769 patent"), entitled "Text encoding using dummy font," issued on Dec. 18, 2007. (See

http://patft.uspto.gov/netacgi/nph-Parser?

Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO

%2Fsrchnum.htm&r=1&f=G&l=50&s1=7,310,769.PN.&OS=PN/7,310,769&RS=

PN/7,310,769.)

463. The '769 patent contains six independent claims.

464. The first independent claim of the '769 patent reads as follows:

> 1. A computer-implemented method for processing an electronic document including a non-coded representation of characters of text, the method comprising: receiving text coding information identifying each of a plurality of characters of text represented by the non-coded representation; based on the text coding information, generating a coded representation that: specifies for each of the plurality of characters of text represented by the non-coded representation, a respective code value based on a character encoding, the code value selected to represent the respective character of text, and associates a particular glyph with all the code

values specified for each of the plurality of characters of text, the particular glyph selected independently of any relation to the respective character of text, the collection of glyphs being insufficient to semantically represent the plurality of characters of text; associating the coded representation with the non-coded representation, wherein a code value of one of the plurality of characters of text is associated with one or more positions in a visual representation of the non-coded representation; displaying the visual representation of the non-coded representation; receiving user input selecting one or more positions in the visual representation; and using the coded representation associated with the non-coded visual representation to identify characters associated with the selected positions of the visual representation.

465. Upon information and belief, claim 1 of the '769 patent does not cover Adobe Reader.

466. Upon information and belief, Defendant has no reasonable basis to believe that claim 1 of the '769 patent covers Adobe Reader.

467. The second independent claim of the '769 patent reads as follows:

15. A system, comprising: one or more computers; a display device; one or more storage devices; a computer program product on the one or more storage devices, the computer program product operable to cause the one or more computers to perform operations comprising: receive text coding information identifying each of a plurality of characters of text represented by a non-coded representation; based on the text coding

information, generate a coded representation that: specifies for each of the plurality of characters of text represented by the non-coded representation, a respective code value based on a character encoding, the code value selected to represent the respective character of text, and associates a particular glyph with all the code values specified for each of the plurality of characters of text, the particular glyph selected independently of any relation to the respective character of text, the collection of glyphs being insufficient to semantically represent the plurality of characters of text; associate the coded representation with the non-coded representation, wherein a code value of one of the plurality of characters of text is associated with one or more positions in a visual representation of the non-coded representation; display the visual representation of the non-coded representation; receive user input selecting one or more positions in the visual representation; and use the coded representation associated with the non-coded visual representation to identify characters associated with the selected positions of the visual representation.

468.   Upon information and belief, claim 15 of the '769 patent does not

cover Adobe Reader.

469.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 15 of the '769 patent covers Adobe Reader.

470.   The third independent claim of the '769 patent reads as follows:

28. A computer-implemented method for processing an electronic document including a non-coded representation of characters of text, the

method comprising: receiving text coding information identifying each of a plurality of characters of text represented by the non-coded representation; based on the text coding information, generating a coded representation that: specifies for each of the plurality of characters of text a respective code value based on a single character encoding, the code value selected to represent the respective character of text, and associates all of the specified code values with a single glyph, the single glyph being insufficient to semantically represent the plurality of characters of text; associating the coded representation with the non-coded representation, wherein a code value of one of the plurality of characters of text is associated with one or more positions in a visual representation of the non-coded representation; displaying the visual representation of the non-coded representation; receiving user input selecting one or more positions in the visual representation; and using the coded representation associated with the non-coded visual representation to identify characters associated with the selected positions of the visual representation.

471.   Upon information and belief, claim 28 of the '769 patent does not cover Adobe Reader.

472.   Upon information and belief, Defendant has no reasonable basis to believe that claim 28 of the '769 patent covers Adobe Reader.

473.   The fourth independent claim of the '769 patent reads as follows:

32. A system, comprising: one or more computers; a display device; one or more storage

devices; a computer program product on the one or more storage devices, the computer program product operable to cause the one or more computers to perform operations comprising: receiving text coding information identifying each of a plurality of characters of text represented by a non-coded representation; based on the text coding information, generating a coded representation that: specifies for each of the plurality of characters of text a respective code value based on a single character encoding, the code value selected to represent the respective character of text, and associates all of the specified code values with a single glyph, the single glyph being insufficient to semantically represent the plurality of characters of text; associating the non-coded representation with the coded representation, wherein a code value of one of the plurality of characters of text is associated with one or more positions in a visual representation of the non-coded representation; displaying the visual representation of the non-coded representation; receiving user input selecting one or more positions in the visual representation; and using the coded representation associated with the non-coded visual representation to identify characters associated with the selected positions of the visual representation.

474.   Upon information and belief, claim 32 of the '769 patent does not cover Adobe Reader.

475.   Upon information and belief, Defendant has no reasonable basis to believe that claim 32 of the '769 patent covers Adobe Reader.

476.   The fifth independent claim of the '769 patent reads as follows:

37. A computer program product, embodied on a machine-readable medium, for processing an electronic document including a non-coded representation of characters of text, the computer program product including instructions operable to cause data processing apparatus to: receive text coding information identifying each of a plurality of characters of text represented by the non-coded representation; based on the text coding information, generate a coded representation that: specifies for each of the plurality of characters of text represented by the non-coded representation, a respective code value based on a character encoding, the code value selected to represent the respective character of text, and associates a particular glyph with all the code values specified for each of the plurality of characters of text, the particular glyph selected independently of any relation to the respective character of text, the collection of glyphs being insufficient to semantically represent the plurality of characters of text; associate the coded representation with the non-coded representation, wherein a code value of one of the plurality of characters of text is associated with one or more positions in a visual representation of the non-coded representation; displaying the visual representation of the non-coded representation; receiving user input selecting one or more positions in the visual representation; and using the coded representation associated with the non-coded visual representation to identify characters associated with the selected positions of the visual representation.

477.   Upon information and belief, claim 37 of the '769 patent does not

cover Adobe Reader.

478.   Upon information and belief, Defendant has no reasonable basis to

believe that claim 37 of the '769 patent covers Adobe Reader.

479.   The final independent claim of the '769 patent reads as follows:

> 51. A computer program product, embodied
> on a machine-readable medium, for processing an
> electronic document including a non-coded
> representation of characters of text, the computer
> program product including instructions operable to
> cause data processing apparatus to: receive text
> coding information identifying each of a plurality
> of characters of text represented by the non-coded
> representation; based on the text coding
> information, generate a coded representation that:
> specifies for each of the plurality of characters of
> text a respective code value based on a single
> character encoding, the code value selected to
> represent the respective character of text, and
> associates all of the specified code values with a
> single glyph, the single glyph being insufficient to
> semantically represent the plurality of characters of
> text; associate the coded representation with the
> non-coded representation, wherein a code value of
> one of the plurality of characters of text is
> associated with one or more positions in a visual
> representation of the non-coded representation;
> display the visual representation of the non-coded
> representation; receive user input selecting one or
> more positions in the visual representation; and
> using the coded representation associated with the
> non-coded visual representation to identify
> characters associated with the selected positions of
> the visual representation.

480.   Upon information and belief, claim 51 of the '769 patent does not

- 136 -

cover Adobe Reader.

481.   Upon information and belief, Defendant has no reasonable basis to believe that claim 51 of the '769 patent covers Adobe Reader.

482.   Because no independent claim of the '769 patent covers Adobe Reader, the '769 patent does not cover Adobe Reader. Moreover, Defendant lacks any reasonable basis to believe that it does.

483.   Upon information and belief, over the past five years and continuing to the present time, Defendant has and continues to mark its Adobe Reader product(s) with the number(s) of one or more additional inapplicable patent(s) (the "Additional Inapplicable Patent(s)") that do not cover Adobe Reader, but whose inapplicability is difficult or impossible to ascertain from observation of the distributed Reader program. Upon information and belief, Defendant lacks any reasonable basis to believe that the Additional Inapplicable Patent(s) cover Adobe Reader and mis-marks the same with intent to deter competition by falsely asserting to the public -- including potential competitors -- that Adobe Reader is protected by the Additional Inapplicable Patent(s), in further violation of 35 U.S.C. §292.

484.   Upon information and belief, Defendant is a large and sophisticated company that regularly employs and retains multiple lawyers familiar with the

requirements of 35 U.S.C. §292.

485.   Upon information and belief, one or more of the following employees of defendant (collectively, the "Compliance Team") has/have responsibility for ensuring that Adobe Reader complies with 35 U.S.C. §292: Curtis Arnold, Mindy Laponis, Rachel Liu, David Nix, James Oh, Jennifer Ruehr, Lynsey Sayers, Ann Sellier, and/or Brigit Young.

486.   Upon information and belief, one or more members of the Compliance Team either create/created, participate/participated in the creation of, and/or approve/approved the the patent marking(s) used in releases of Adobe Reader.

487.   Upon information and belief, those members of the Compliance Team responsible for §292 compliance either (i) know that the Inapplicable Patents do not cover Adobe Reader or (ii) have no reasonable basis to believe that the Inapplicable Patents cover Adobe Reader.

488.   Nevertheless, upon information and belief, those members of the Compliance Team responsible for §292 compliance have allowed, and continue to allow, mis-marking of the Inapplicable Patents despite actual knowledge that such mis-marking violates federal law.

489.   Indeed, after receiving specific notice, on or about Dec. 30, 2009,

that the legality of Adobe Reader's patent notices had been called into question by the SF Tech. lawsuit, members of the Compliance Team made no substantial effort to correct even the specific deficiencies noted in the SF Tech. complaint (i.e., the expired patents), much less to rectify the larger problem involving the numerous Inapplicable Patents. Accordingly, since at least early January, 2010, Defendant's marking of expired and inapplicable patents on Adobe Reader has been deliberate and willful, thus making this case exceptional.

490.   Upon information and belief, one motivation for Defendant's mis-marking of its Reader product with numerous expired and inapplicable patents is to deter members of the public, such as independent software developers, from implementing competing products that read and/or write PDF-format files. Specifically, because a reasonable member of the public would be justified in assuming that Defendant's patent markings are correct, such member of the public would need to unnecessarily evaluate more than thirty irrelevant patents as part of any responsible effort to implement competing PDF-enabled software. Upon information and belief, the cost to such member of the public to have the Inapplicable Patents evaluated by a competent patent attorney would run well in excess of $30,000.00.

491.   Defendant, upon information and belief, is well aware of the large

costs associated with a potential competitor's evaluation of the Inapplicable

Patents, and Defendant's desire to burden potential competitors with such costs is

one of the reasons why Defendant has and continues to mis-mark its Adobe Reader

product as "[p]rotected by" the Inapplicable Patents.

492.   Moreover, Defendant's web site (see

http://partners.adobe.com/public/developer/support/topic_legal_notices.html)

contains the following warning concerning the limited scope of the public's right to

use patented Adobe technology in connection with the reading and/or writing of

PDF-format files:

> **Adobe Patent Clarification Notice:**
> **Reading and writing PDF files**
>
> Adobe has a number of patents covering
> technology that is disclosed in the Portable
> Document Format (PDF) Specification, version 1.3
> and later, as documented in PDF Reference and
> associated Technical Notes (the "Specification").
> Adobe desires to promote the use of PDF for
> information interchange among diverse products
> and applications. Accordingly, the following
> patents are licensed on a royalty-free, nonexclusive
> basis for the term of each patent and for the sole
> purpose of developing software that produces,
> consumes, and interprets PDF files that are
> compliant with the Specification:
>
> U.S. Patent Numbers:
>
>   * 5,634,064

* 5,737,599
* 5,781,785
* 5,819,301
* 6,028,583
* 6,289,364
* 6,421,460

In addition, the following patent is licensed on a royalty-free, nonexclusive basis for its term and for the sole purpose of developing software that produces PDF files that are compliant with the Specification (specifically excluding, however, software that consumes and/or interprets PDF files):

U.S. Patent Number:

* 5,860,074

... The above licenses are limited to only those rights required to implement the Specification and no others. That is to say, Adobe grants only those rights in the above patent(s) necessarily practiced to implement the Specification, and does not grant any rights not required to implement the Specification. The licenses do not grant the right to practice any patent covering other technologies, such as implementation techniques that are not explicitly disclosed in the Specification, nor does it allow the use of any patented feature for any purpose other than as set forth in the applicable license grant. Adobe has other patents in various fields, none of which are hereby licensed.

493.   Upon information and belief, the above "Patent Clarification Notice"

is designed to, and does, enhance the competitive deterrence effect of Defendant's

mis-marking of Adobe Reader products with numerous expired and inapplicable patents.

494.   Each falsely-marked patent number constitutes a separate "offense" under 35 U.S.C. §292. Thus, for example, each installation of Adobe Reader 9.3.1 on a computer in the United States represents no less than thirty-five "offenses" under §292 (four for the expired patents and at least thirty-one for the inapplicable patents).

495.   Defendant's violation of 35 U.S.C. §292(a) injures the sovereign interest of the United States.

496.   Defendant's violation of 35 U.S.C. §292(a) injures the general public interested represented by PUBPAT, such as by disseminating false information to consumers and decreasing competition.

PRAYER FOR RELIEF

WHEREFORE, plaintiff PUBPAT respectfully requests that this Court:

(1)   Find that Defendant's distribution and/or sale of knowingly mislabeled products violates 35 U.S.C. §292(a);

(2)   Determine an appropriate "fine," not more than $500 per offense, but sufficient to appropriately penalize Defendant's willful violations of §292(a), and

to deter Defendant and others similarly situated from violating §292(a) in the

future;

    (3)   Direct that half of the fine be paid to the United States government;

and,

    (4)   Direct that the other half of the fine be paid to PUBPAT.

            Respectfully submitted,

            PUBLIC PATENT FOUNDATION, INC.

By: _Daniel B. Ravicher_

Dated: New York, New York   Daniel Ravicher, Esq. (DR1498)
      March 9, 2010   David Garrod, Ph.D., Esq. (DG6759)
            PUBLIC PATENT FOUNDATION, INC.
            Benjamin N. Cardozo School of Law
            55 Fifth Avenue
            New York, New York 10003
            Tel.: 212-545-5337
            Fax.: 212-591-6038